PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-249694

(43)Date of publication of application : 14.09.2001

(51)Int.Cl. G10L 19/00
G06K 17/00
G06K 19/07
G06K 19/00
G10K 15/02
G11C 7/00

(21)Application number : 2000-156755

(22)Date of filing : 26.05.2000

(71)Applicant : MATSUSHITA ELECTRIC IND CO LTD

(72)Inventor : TAGAWA KENJI
HIROTA TERUTO
ISHIKAWA TOMOKAZU
INOUE SHINJI
MATSUSHIMA HIDEKI
KOZUKA MASAYUKI

(30)Priority

Priority number	Priority date	Priority country
11149893	28.05.1999	JP
11236724	24.08.1999	JP
11372605	28.12.1999	JP

(54) SEMICONDUCTOR MEMORY CARD, REPRODUCING DEVICE, RECORDING DEVICE, REPRODUCING METHOD, RECORDING METHOD, AND COMPUTER-READABLE RECORDING MEDIUM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a semiconductor memory card permitting to avoid reproducing an overlapping part without specifying a reproducing position again in the device, when a music album reproduced by one reproducing device is reproduced by another reproducing device.

SOLUTION: The semiconductor device stores plural AOB composing plural tracks, and PlayList information showing order of reproduction concerning these tracks, and stores Playlist-Number showing the PlayList information used just before the preceding reproduction, Track-Number showing a reproduced track, and Playback-Time showing the point just before the preceding reproduction by a relative time with respect to the head of the audio block as resume information (PLMG-RSM-PL).

LEGAL STATUS

[Date of request for examination]	26.05.2000
[Date of sending the examiner's decision of rejection]	
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]	
[Date of final disposal for application]	
[Patent number]	3366896
[Date of registration]	01.11.2002
[Number of appeal against examiner's decision of rejection]	
[Date of requesting appeal against examiner's decision of rejection]	

[Date of extinction of right]

CLAIMS

[Claim(s)]

[Claim 1] The semi-conductor memory card characterized by storing the audio sequence which comes to arrange two or more audio objects, and the resume information which shows the restart location in the case of resuming playback from the middle of an audio sequence.

[Claim 2] For the 1st positional information, said resume information is a semi-conductor memory card according to claim 1 characterized by showing the 1st restart location set up through user operation, and the 2nd positional information showing the 2nd restart location automatically set up at the time of the last playback halt including the 1st positional information and/or the 2nd positional information.

[Claim 3] In each of two or more audio objects which can be set to said audio sequence The identification information of a proper is given. Said 1st positional information The identification information given to any one audio object shows the 1st restart location in an audio sequence. Said 2nd positional information The semi-conductor memory card according to claim 2 characterized by showing the 2nd restart location in an audio sequence by the identification information of any one audio object, and the hour entry which shows the offset from the head of the audio object to the 2nd restart location.

[Claim 4] To a semi-conductor memory card, further each salvage pathway information The salvage pathway information on plurality which the identification information given to each of two or more audio object and the playback ranking about each audio object are made to correspond, and shows them, Said resume information contains the specific information which specifies any they are among further two or more salvage pathway information. Said 1st positional information and 2nd positional information The semi-conductor memory card according to claim 3 characterized by showing

the 1st and 2nd restart location in an audio sequence by the identification information about an audio object, and the playback ranking about the audio object in the specified salvage pathway information.

[Claim 5] Said semi-conductor memory card includes two or more subresume information matched with each salvage pathway information. Each ** resume information When two or more audio objects are reproduced based on two or more playback ranking included in salvage pathway information, The positional information included in said resume information including the positional information which shows whether it should reproduce from a location in the middle of which [of which audio object] The semi-conductor memory card according to claim 4 characterized by showing the location as a restart location in the middle of [same] the same audio object shown by the positional information included in which thing among two or more subresume information.

[Claim 6] Said subresume information is a semi-conductor memory card according to claim 5 characterized by being set as the 2nd value if playback of two or more audio object using two or more playback ranking shown in corresponding salvage pathway information is completed, it will be set as the 1st value and playback of two or more audio object using two or more playback ranking will not be completed.

[Claim 7] The audio sequence which comes to arrange two or more audio objects, It is a regenerative apparatus about the semi-conductor memory card which stores the resume information which shows the restart location in the case of resuming playback from the middle of an audio sequence. The 1st playback actuation which specified which audio object, Or a reception means to receive from an operator the 2nd playback actuation of specifying neither of the audio objects, When the audio object specified among audio sequences when the 1st playback actuation was received is reproduced and the 2nd playback actuation is received, resume information is read from a semi-conductor memory card. The regenerative apparatus characterized by having a playback means to resume playback of an audio sequence from the restart location shown in resume information in the audio sequence.

[Claim 8] It is the regenerative apparatus according to claim 7 characterized

by to perform regeneration from a location while being shown in a hour entry in the audio object shown in the identification information contained in resume information when the restart location in an audio sequence is shown and said playback means receives the 2nd playback actuation by the hour entry said resume information indicates the offset from the identification information and the head of an audio object of any one audio object to a restart location to be.

[Claim 9] It is the regenerative apparatus about a semi-conductor memory card with which the audio sequence containing two or more audio objects and the 1st resume information which shows the restart location set up through user operation are stored. A judgment means to judge whether it is justly written in the semi-conductor memory card loaded with the 2nd resume information which indicates the restart location automatically set up at the time of the last playback halt to be the charger stage which loads with a semi-conductor memory card, When an audio sequence is reproduced based on the 2nd resume information when the 2nd resume information is written in justly, and not written in justly, the 1st resume information is read from a semi-conductor memory card. The regenerative apparatus characterized by having a playback means to reproduce an audio sequence based on this 1st resume information.

[Claim 10] It is the regenerative apparatus according to claim 9 which said regenerative apparatus is equipped with a storage means memorize the flag which shows whether the 2nd resume information is used further or the 1st resume information uses, and carries out [that it is reproduced based on the 1st resume information even if said playback means is the case the 2nd resume information is justly written in the semi-conductor memory card with which it was loaded when the purport for which a flag uses the 1st resume information was shown, and put in, and] as the description.

[Claim 11] Said regenerative apparatus is a regenerative apparatus according to claim 10 characterized by having a reception means to receive from an operator the actuation which shows assignment of any to use between the 2nd resume information and the 1st resume information further, and a setting-out means to set up the flag with which the storage means has memorized according to the actuation which the reception means received.

[Claim 12] A reception means to be a recording device about a semi-conductor memory card, and to receive the actuation from an operator, The playback means which carries out sequential playback of the audio object contained in an audio sequence when the received actuation is playback actuation, When the received actuation is halt actuation, it is based on the playback location currently reproduced. The recording device characterized by having a record means to record the resume information which pinpoints the restart location in the case of resuming playback from the middle of an audio sequence, and shows the restart location concerned on a semi-conductor memory card.

[Claim 13] The audio sequence which comes to arrange two or more audio objects, It is the record medium which is recording the program to which the playback procedure about the semi-conductor memory card which stores the resume information which shows the restart location in the case of resuming playback from the middle of an audio sequence is made to carry out to a computer in the format in which computer reading is possible. The 1st playback actuation which specified which audio object, Or the reception step which receives from an operator the 2nd playback actuation of specifying neither of the audio objects, When the audio object specified among audio sequences when the 1st playback actuation was received is reproduced and the 2nd playback actuation is received, resume information is read from a semi-conductor memory card. The record medium which is characterized by recording the ****** program on the computer in the procedure which consists of a playback step which resumes playback of an audio sequence from the restart location shown in resume information in the audio sequence and in which computer reading is possible.

[Claim 14] Said resume information by the identification information of any one audio object, and the hour entry which shows the offset from the head of the audio object to a restart location The restart location in an audio sequence is shown. Said playback step The record medium which is characterized by performing regeneration from a location while being shown in a hour entry in the audio object shown in the identification information contained in resume information when the 2nd playback actuation is received and in which

computer reading according to claim 13 is possible.

[Claim 15] The audio sequence containing two or more audio objects, It is the record medium which is recording the program to which the playback procedure about a semi-conductor memory card in which the 1st resume information which shows the restart location set up through user operation is stored is made to carry out to a computer in the format in which computer reading is possible. The judgment step which judges whether it is justly written in the semi-conductor memory card loaded with the 2nd resume information which indicates the restart location automatically set up at the time of the last playback halt to be the loading step which loads with a semi-conductor memory card, When an audio sequence is reproduced based on the 2nd resume information when the 2nd resume information is written in justly, and not written in justly, the 1st resume information is read from a semi-conductor memory card. The record medium which is characterized by recording the ****** program on the computer in the procedure which consists of a playback step which reproduces an audio sequence based on this 1st resume information and in which computer reading is possible.

[Claim 16] Said computer is further equipped with a storage means to memorize the flag which shows whether the 2nd resume information is used or the 1st resume information is used. Said playback step When the flag shows the purport using the 1st resume information, even if it is the case where the 2nd resume information is justly written in the semi-conductor memory card with which it was loaded, and it puts in The record medium which is characterized by reproducing based on the 1st resume information and in which computer reading according to claim 15 is possible.

[Claim 17] Said program is a record medium which is characterized by to consist of a reception step which receives from an operator the actuation which shows assignment of any to use between the 2nd resume information and the 1st resume information further, and a setting-out step which sets up the flag with which the storage means has memorized according to the actuation which the reception step received and in which computer reading according to claim 16 is possible.

[Claim 18] The reception step which is the record medium which is recording

the program to which the record procedure about a semi-conductor memory card is made to carry out to a computer in the format in which computer reading is possible, and receives the actuation from an operator, The playback step which carries out sequential playback of the audio object contained in an audio sequence when the received actuation is playback actuation, When the received actuation is halt actuation, it is based on the playback location currently reproduced when halt actuation was performed. The restart location in the case of resuming playback from the middle of an audio sequence is pinpointed. The record medium which is characterized by recording the ****** program on the computer in the procedure which consists of a record step which records the resume information which shows the restart location concerned on a semi-conductor memory card and in which computer reading is possible.

[Claim 19] The audio sequence which comes to arrange two or more audio objects, It is the playback approach about the semi-conductor memory card which stores the resume information which shows the restart location in the case of resuming playback from the middle of an audio sequence. The 1st playback actuation which specified which audio object, Or the reception step which receives from an operator the 2nd playback actuation of specifying neither of the audio objects, When the audio object specified among audio sequences when the 1st playback actuation was received is reproduced and the 2nd playback actuation is received, resume information is read from a semi-conductor memory card. The playback approach characterized by consisting of a playback step which resumes playback of an audio sequence from the restart location shown in resume information in the audio sequence.

[Claim 20] Said resume information by the identification information of any one audio object, and the hour entry which shows the offset from the head of the audio object to a restart location The restart location in an audio sequence is shown. Said playback step The playback approach according to claim 19 characterized by performing regeneration from a location while being shown in a hour entry in the audio object shown in the identification information contained in resume information when the 2nd playback actuation is received.

[Claim 21] It is the playback approach applied to the regenerative apparatus

about a semi-conductor memory card with which the audio sequence containing two or more audio objects and the 1st resume information which shows the restart location set up through user operation are stored. The judgment step which judges whether it is justly written in the semi-conductor memory card loaded with the 2nd resume information which indicates the restart location automatically set up at the time of the last playback halt to be the loading step which loads with a semi-conductor memory card, When an audio sequence is reproduced based on the 2nd resume information when the 2nd resume information is written in justly, and not written in justly, the 1st resume information is read from a semi-conductor memory card. The playback approach characterized by consisting of a playback step which reproduces an audio sequence based on this 1st resume information.

[Claim 22] It is the playback approach according to claim 21 which said regenerative apparatus is equipped with a storage means memorize the flag which shows whether the 2nd resume information uses further or the 1st resume information uses, and carries out [that it is reproduced based on the 1st resume information even if said playback step is the case the 2nd resume information is justly written in the semi-conductor memory card with which it was loaded when the purport for which a flag uses the 1st resume information was shown, and put in, and] as the description.

[Claim 23] The reception step which is the record approach about a semi-conductor memory card, and receives the actuation from an operator, The playback step which carries out sequential playback of the audio object contained in an audio sequence when the received actuation is playback actuation, When the received actuation is halt actuation, it is based on the playback location currently reproduced when halt actuation was performed. The record approach characterized by consisting of a record step which records the resume information which pinpoints the restart location in the case of resuming playback from the middle of an audio sequence, and shows the restart location concerned on a semi-conductor memory card.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to the amelioration in the case of storing especially the audio data and the control data which were distributed as contents in contents distribution services, such as an electronic music distribution, about the semi-conductor memory card which stores audio data and control data, a regenerative apparatus, a recording device, the playback approach, the record approach, and the record medium in which computer reading is possible.

[0002]

[Description of the Prior Art] The electronic music distribution whose purchase of a music content is attained in the Internet can become the explosion material of activation of a music commercial scene, and the infrastructure for the implementation is being fixed steadily. The semi-conductor memory card mentioned above stores the music content purchased in the electronic music distribution, it is the record medium of a suitable portable mold to carry this, and the need will be expected to increase by leaps and bounds from now on.

[0003] There are various classes of semi-conductor memory cards, such as a flash plate ATA card and a CompactFlash (trademark) card, and the thing of disk molds, such as CD-R and a mini disc (MD), can also be used for record of a music content besides a semi-conductor memory card. The approach of specifying where various things existing in the record medium for storing of a music content reproducing the music content (music) recorded on these record media from although many need people and a dealings person are just going to recognize, and the specification method of the so-called playback part do not necessarily have so many classes, and are extracted to that pattern how many kinds.

[0004] If instantiation listing of the thing typical about the specification method of the playback part in the case of reproducing the music album containing two or more music contents (music) is carried out, the input of the thing (1) of

reproducing music from a top thing among two or more music, and a tune number number will be received from an operator, and the thing (2) of making playback start from the music to which the tune number number was given will be mentioned. if the specification method of the playback part of these (1) - (2) is analyzed, with a specification method (1), playback will always be started from the same music and a user will listen to the music contained in a music album in the same sequence from a head -- things -- ** Here, since a regenerative apparatus starts playback from the music of the head of a music album in case playback is stopped and the music album is again reproduced after hearing a music album to the middle, an operator puts up with the music listened to once, and has to continue hearing it.

[0005] In case playback is stopped and the music album is again reproduced from the music specified by an operator in a specification method (2) after hearing a music album to the middle since playback is started, by inputting into a regenerative apparatus the tune number number of the music which should be reproduced next, an operator does not need to reproduce a music album, and does not need to put up with and listen to the music listened to once from the music or subsequent ones. However, in this case, a user will have to operate the input of a tune number number etc. and will trouble excessive time and effort to a user. Moreover, the music from which thinking that the music which should hear it after this when the operator does not remember to accuracy which music exists in what No. was specified differs is specified, and it is also possible to reproduce the mistaken music.

[0006] As mentioned above, by specification method (1) - (2), when resuming playback after hearing a music album to the middle, the music listened to once must be put up with and listened to, or alter operation of a tune number number must be performed, and it is hard to say that it is user FURENDORII at this point. (1) to the specification method of playback locations other than (2) The music which should start playback by forward-search playback or hard flow search playback, Although the music assignment using the thing (3) of specifying the playback event in the music, a jog dial, etc., and assignment of playback start time are received from an operator and there is a thing (4) of starting playback etc., from the music and playback start time which were

specified In the point of making an operator specify the location which playback completed until now, it can be said that these have the same trouble as a specification method (2). the specification method of these (1) - (4) -- comparing -- more -- a user -- the specification method of the playback approach in the regenerative apparatus (generally called MD player) of the present mini disc is in the specification method of a FURENDORII playback part.

[0007] This specification method will be a thing of making playback of the music album currently recorded on MD resume according to the resume information concerned, if it is made to store in the nonvolatile memory in which the resume information which shows that halt event is contained by MD player and playback of the MD concerned is again directed, when a mini disc (MD) is played and that playback is suspended. In this specification method, even after the power source of MD player is severed, resume information is maintained, without being eliminated. Therefore, after hearing a music album to the middle, playback is suspended, and even if it is the case where a power source is severed, a music album can be reproduced from immediately after the location reproduced last time. Under the present circumstances, since the playback from the head of a music album is not repeated repeatedly and it does not trouble to the input of a tune number number like a specification method (2) like a specification method (1), it is the optimal when appreciating the music album containing two or more music.

[0008]

[Problem(s) to be Solved by the Invention] by the way, the case where took out MD from MD player and another MD player is loaded since the resume information which shows how far it was reproduced in the case of MD was memorized by the hardware of MD player -- another MD player concerned -- a specification method (1) -- there is a trouble of reproducing from the music of the beginning of the MD concerned, similarly. Since the another regenerative apparatus concerned has not memorized the resume information which shows the last halt location in case stopping playback and reproducing the music album in another regenerative apparatus, speaking concretely, after hearing a music album to the middle with a certain regenerative apparatus,

the music which the operator listened to [starting playback and] once is put up with, and hearing it from the music of the head of a music album must be continued.

[0009] But if it is rare to hear the music album heard with a certain regenerative apparatus with another regenerative apparatus, he can think that redo of the above playbacks is also considered that it cannot hardly generate, and this is not regarded as questionable, but when it is the music album with which the music album which should be recorded on a record medium was distributed in the electronic music distribution, there is a possibility that hearing the music album heard with a certain regenerative apparatus with another regenerative apparatus may occur frequently.

[0010] The turnover of the music album in an electronic music distribution is realized when the computer which a consumer owns downloads a music album from the server computer of a concert company. In this way, when a music album is downloaded, an operator may reproduce a music album with the general-purpose personal computer. This is because the general-purpose personal computer in recent years has the ability to regenerate of a suitable music content, so an operator is going to try listening the purchased music album using this. The same music album is recorded on a record medium, and suppose that the music album was reproduced with the pocket mold regenerative apparatus, after making a general-purpose personal computer reproduce a music album such.

[0011] In this case, in a general-purpose personal computer, since a pocket mold regenerative apparatus cannot know how far the music album was reproduced, a pocket mold regenerative apparatus will reproduce the same music album from the beginning. Such, if the music album was reproduced from the beginning, the music of the music album once heard with the general-purpose personal computer will have to be listened to again, and an operator will get bored with the repeat of the same playback of music.

[0012] The music album which consists of a huge number of music is recorded on one record medium with large-capacity[the miniaturization of a record medium, lightweight-izing, and]-izing, and it is thought that reproducing this with various regenerative apparatus will be performed

frequently from now on. In this case, although it is thought that it may happen frequently to hear the music album heard with a certain regenerative apparatus with another regenerative apparatus (it is called transition of a regenerative apparatus to make it reproduce with another regenerative apparatus after reproducing a music album with a certain regenerative apparatus), by making it reproduce, it can never be said from the beginning for an operator that the music album which contains huge music in whenever [that] is desirable.

[0013] The 1st object of this invention is offering the semi-conductor memory card which can avoid playback of a duplication part, without making a playback location specify anew in the equipment, when reproducing the music album heard with a certain regenerative apparatus with another regenerative apparatus. The 2nd object of this invention is offering the semi-conductor memory card which a regenerative apparatus's is made to reproduce, without overlapping and reproducing the content reproduced once, when reproducing the music album heard with a certain regenerative apparatus with another regenerative apparatus.

[0014]

[Means for Solving the Problem] The 1st object of the above is attained by the semi-conductor memory card characterized by storing the audio sequence which comes to arrange two or more audio objects, and the resume information which shows the restart location in the case of resuming playback from the middle of an audio sequence.

[0015]

[Embodiment of the Invention] Henceforth, the operation gestalt of a semi-conductor memory card (flash memory card) is explained, referring to a drawing. In addition, the class number which has the following systems in the beginning of a sentence is given to each subsequent sentence.

[0016] The digit count of {x1-x2_x3-x4} class number means the hierarchical depth of the item. Speaking concretely, x1 being a drawing number quoted to explanation. Since the number in alignment with the sequence quoted in a description is given to drawing attached to this description, the sequence of this drawing number becomes almost the same as that of the sequence of

explanation. x2 shows the sequence of explanation in the case of quoting and explaining drawing shown in x1. In order that x3 may explain the component of x2 to a detail more, when quoting an explanatory view, the drawing number of the explanatory view is shown and x4 shows the sequence of explanation in the case of quoting and explaining drawing shown in xthree.

[0017] (The 1st operation gestalt)

{1-1_2} The appearance configuration of flash memory card 31 is explained at the beginning of the appearance configuration of flash memory card 31. Drawing 1 is a configuration **** Fig. at the time of seeing flash memory card 31 from a top face, and drawing 2 is drawing showing the structure at the time of seeing flash memory card 31 from the underside. As shown in drawing 1 and drawing 2 , the die length of the magnitude of flash memory card 31 is the magnitude (magnitude of stamp size) of extent which about 32.0mm and width of face are about 24.0 mm, and thickness about 2.0 mm, and can grasp them by the fingertip. Nine connectors for connection with a device are prepared in the underside, and the protection switch 32 which an operator can set up is formed [to which overwrite of the content of storage is permitted / or or / whether prohibition is carried out and] in the side face.

[0018] {3-1} Physical structure drawing 3 of flash memory card 31 is drawing showing the layered structure of the semi-conductor memory card (flash memory card 31 is called hereafter) concerning this operation gestalt. Although the layered structure of flash memory card 31 is the point which consists of the physical layer, a file system layer, and an application layer and is the same as the layered structure of DVD (Digital Video Disc) as shown in this Fig., the logical structure in each class and the physical structure are greatly different.

[0019] {3-2} physical **** of flash memory card 31 -- the physical layer of flash memory card 31 is explained first. A flash memory consists of two or more sectors, and each sector stores 512 bytes of digital data. For example, in the case of the 64MByte type flash memory card 31, that memory capacity is 67108864 (= 64x1024x1024) cutting tools, and the number of effective sectors at this time is set to 131072 (= 67108864/512). furthermore, if the number of alternate sectors for an error is deducted from this effective sector, the

remaining numbers of effective sectors will be set to 128,000, and various data will be recorded here -- things -- **

[0020] {3-2_4A-1} three fields in the physical layer -- three fields shown in drawing 4 (a) are established in the field which consists of these effective sectors. Drawing 4 (a) is drawing showing the "system area", the "protection field", and the "user data area" which were established in the physical layer of flash memory card 31. Henceforth, these three fields are explained.

[0021] The device by which the "user data area" was connected with flash memory card 31 can write in various data freely, it is the field which can read data freely and the contrant region is managed by the file system. A "system area" is a field where the media ID with a value unique about each of flash memory card 31 are stored. To the ability to write in a user data area, a system area is a read only and cannot rewrite the media ID stored here.

[0022] A "protection field" is a field in which data writing is possible as well as a user data area. The difference with a user data area is a point whose R/W is attained, only when mutual recognition with the point which can be written only when justification with the device connected with flash memory card 31 and flash memory card 31 mutual in a protection field is checked to the ability to write data freely in a user data area, i.e., the device connected with flash memory card 31, and flash memory card 31 is successful.

[0023] {3-2_4A-2} In case the device connected to the application flash memory card 31 of three fields in the physical layer writes data in flash memory card 31, these three fields are used according to the necessity of the protection of copyrights of the data. Here, when writing data to be protected [of copyright] in flash memory card 31, the data concerned are stored in a user data area after being enciphered using a predetermined cryptographic key (referred to as FileKey.). A copyright person can set up this FileKey freely, and it enciphers the FileKey itself used for this encryption in order to take all possible measures further, although the copyright of this and the data concerned is protected. In case the FileKey itself is enciphered, the any value obtained by applying the media ID stored in the system area to predetermined operation expression is used as a key, and a protection field stores FileKey enciphered using the any value concerned. Since two steps of encryption of

enciphering the data which need protection of copyrights using predetermined FileKey, and enciphering this FileKey itself using the value based on Media ID is made, literary piracy acts, such as an illegal copy, become very difficult.

[0024] {3-2_4B-1} The configuration of the physical layer of the outline flash memory card 31 of a file system is as having explained above, and it turns out that amelioration of protection of copyrights is made. Then, the configuration of the file system layer which exists on this physical layer is explained. the file system layer of DVD is the file system of a UDF (universal disk format) mold -- it receives, and the file system layer of flash memory card 31 is the file system (FAT:File Allocation Table, ISO/IEC9293) of a FAT mold, and this point differs from DVD.

[0025] Drawing 4 (b) is drawing showing the configuration of the protection field in a file system layer, and a user data area. It is clear also from this drawing that the protection field and user data area in a file system include the "data area" with the "partition boot sector", and "a file allocation table (FAT)" and a "root directory entry", and have the composition with both the same protection fields and user data areas in drawing 4 (b). Drawing 5 is drawing showing the detail of these file system organizations. Henceforth, the configuration about a user data area is explained, referring to drawing 4 and drawing 5 .

[0026] {3-2_4B-2} A partition boot sector "a partition boot sector" is a sector the content which a general-purpose personal computer should refer to at the time of boot is indicated to be, when a general-purpose personal computer is loaded with flash memory card 31 and it is able to assign flash memory card 31 to the startup disk of the operating system of the general-purpose personal computer concerned.

[0027] {3-2_4B-3_5} A data area "a data area" is a field accessed by the device which made the cluster the smallest unit and was connected to flash memory card 31. To the sector size of flash memory card 31 being 512 bytes, since cluster size is 16 K bytes, in a file system layer, R/W of data is performed by making 32 sectors into one unit. The reason for having made cluster size into 16 K bytes is as follows. That is, data writing must be performed once it erases the data stored in the flash memory card 31

concerned (elimination), when writing data in flash memory card 31. In flash memory card 31, since the size which can erase data such is 16 K bytes, data writing is made to be performed by setting cluster size as the size in which this erasion is possible suitably. two or more clusters 002, 003, 004, and 005 in which the outgoing line ff2 of the broken line in drawing 5 is contained in a data area ..... is shown. the numbers 002, 003, 004, 005, 006, 007, and 008 in drawing -- the cluster number of the hexadecimal notation of triple figures given in order that ..... might identify each cluster is shown. Since access to a data area is performed considering a cluster as a smallest unit, the internal location of a data area is directed using these cluster numbers.

[0028] {3-2_4B-4_5} The file allocation table "a file allocation table" has the file system organization based on ISO/IEC 9293, and consists of two or more FAT values. It is shown which cluster each FAT value should just read next, when reading appearance of the cluster which is matched with each cluster and corresponds is carried out. two or more FAT values 002, 003, 004, and 005 by which the outgoing line ff1 of the broken line of drawing 5 is contained in a file allocation table .. is shown. The numeric value "002, 003, 004, 005 .." given to this FAT value shows the cluster number of the cluster with which each FAT value is matched with which cluster, or [that is,] each FAT value is matched.

[0029] {3-2_4B-5_5-1} A root directory entry "a root directory entry" is information which shows what kind of file exists in a root directory. Speaking concretely, indicating the "cluster number of the file beginning" in which a "file attribute", "the modification time and the date" of a file, and the head section of a file are stored with "the "file name" and the extension of a file" of the file which exists in a root directory entry.

[0030] {3-2_4B-5_5-2} Although the information about the file in the directory entry root directory of a subdirectory is indicated by this root directory entry, the information about a subdirectory is not indicated by this root directory entry. The directory entry about a subdirectory is created in a data area. The SD_Audio directory entry indicated in the data area of drawing 5 is an example of the directory entry about a subdirectory, and "the cluster number of the file beginning" in which a "file attribute", "the modification time and the

date" of a file, and the head section of a file are stored with "the "file name" and the extension of a file" of the file which exists in the subdirectory is described by this SD_Audio directory entry like a root directory entry.

[0031] {3-2_4B-5_6-1} the storing method of an AOB file -- when it stores a file called AOB001.SA1 in an SD_Audio directory here, AOB001.SA1 is stored how, or an example of a file storing method is explained, referring to drawing 6 . Since the minimum access unit of a data area is a cluster as mentioned above, AOB001.SA1 must make cluster size a smallest unit, and must store it in a data area. AOB001.SA1 is first divided into cluster size, and is written in each cluster. Drawing 6 is drawing supposing the condition of dividing AOB001.SA1 into five in all at cluster size, and storing each division part in Clusters 003, 004, 005, 00A, and 00C.

[0032] {3-2_4B-5_7-1} If division storing of storing method AOB001.SA1 of an AOB file is carried out, a directory entry and a file allocation table must be set up like drawing 7 . Drawing 7 is drawing showing a directory entry in case AOB001.SA1 is recorded on two or more clusters, and the example of setting out about a file allocation table. When the head part of AOB001.SA1 is recorded on the cluster 003 in this Fig., the cluster number 003 about the cluster by which the head part is stored in the "first cluster number" in an SD_Audio directory entry is indicated. Henceforth, it turns out that the part which AOB001.SA1 follows is stored in a cluster 004 and a cluster 005. Although the FAT value 003 (004) supports the cluster 003 which stores the head part of AOB001.SA1, this FAT value shows the cluster 004 which stores the part which an AOB file follows. Moreover, although the FAT value 004 (005) and the FAT value 005 (00A) support the clusters 004 and 005 which store the part which is following this, the FAT value of this shows the clusters 005 and 00A which store the part which the degree of an AOB file follows.

[0033] the cluster number indicated by these FAT value -- arrow heads fk1, fk2, fk3, fk4, and fk5 -- it is shown in ..... as -- one by one -- reading -- **** -- if it dies, all the division parts of AOB001.SA1 can be read. The above explanation shows that the data area of flash memory card 31 is accessed considering a cluster as a smallest unit, and the FAT value is matched with each cluster, respectively. In addition, the cluster number "FFF" which shows

that the cluster stores the last part of a file is described by the FAT value matched with the cluster (an example of drawing 7 cluster 00C) which stored the part of the tail of an AOB file.

[0034] The configuration of the application layer which exists on the file system which finished the explanation about the file system of the flash memory card 31 of this invention, then was mentioned above is explained.

{3-3} The outline of the application layer in the outline flash memory card 31 of the application layer in flash memory card 31 is as having been indicated by drawing 3 . As shown in outgoing-line PN1 of the broken line in drawing 3 , the application layer in flash memory card 31 consists of presentation data and navigation data for controlling playback of presentation data.

[0035] As shown in outgoing-line PN2 of the broken line of this Fig., navigation data contain a play list manager (PlaylistManager (PLMG)) and a truck manager (Track Manager (TKMG)) including the audio object group (AOB group) obtained when presentation data encoded voice data, such as music.

{3-3_8A, B-1} The directory block diagram 8 (a) and (b) are drawings showing what kind of file is created by the subordinate of the directory concerned by what kind of directory being constituted by a user data area and the protection field in a file system layer, when it stores these [in an application layer] two data. the file navigation data, such as a play list manager (PlaylistManager (PLMG)) and a truck manager (Track Manager (TKMG)), were mentioned in whose "SD_AUDIO.PLM" in this Fig. and "SD_AUDIO.TKM" -- it is -- "AOB001.SA1", "AOB002.SA1", "AOB003.SA1", and "AOB004.SA1" -- it is the file (henceforth an AOB file) which stored the audio object whose ..... is presentation data.

[0036] The extension "SA" in "AOB0xx.SA1" is the abbreviation for "Secure Audio", and it is shown that these contents of storing have the need for protection of copyrights (in addition, although only eight AOB files are described by drawing 8 (a), this is a mere example and, as for a SD_Audio directory, an AOB file can be stored a maximum of 999 pieces.). Thus, when the need for protection of copyrights is in presentation data, the subdirectory of a name called an SD_Audio directory is prepared in a protection field, and

cryptographic key storing file AOBSA1.KEY is created by the subordinate of the SD_Audio directory. Drawing 8 (b) is drawing showing cryptographic key storing file AOBSA1.KEY stored under SD_Audio. FileKey#1-#8 which are the cryptographic key train which comes to arrange two or more cryptographic keys FileKey in predetermined sequence are stored in cryptographic key storing file AOBSA1.KEY.

[0037] If the server computer of a concert company holds the SD_Audio directory shown in this drawing 8 (a) and (b) and the purchase demand of the music content concerned is emitted by the consumer in an electronic music distribution, this SD_Audio directory is compressed, and after enciphering, the SD_Audio directory which the consumer who emitted the purchase demand owns will be transmitted through a public network. If the computer which a consumer owns receives this SD_Audio directory, while canceling encryption of this directory, it elongates and a SD_Audio directory is obtained (in addition, a public network here includes all the networks where utilization is released by the public for wire nets, such as an ISDN circuit, the radiocommunication network represented by the cellular phone). In addition, an AOB file may be downloaded from the server computer of a concert company, and the computer which a consumer owns may create the SD_Audio directory shown in drawing 8 (a) of a flash-memory-card 31 odor lever, and (b).

[0038] {3-3_9-1} Response drawing 9 of AOBSA1.KEY and an AOB file is AOBSA1.KEY under SD_Audio, and drawing showing a response with an AOB file. FileKey used when enciphering the encryption file in a user data area in this Fig. is stored in the cryptographic key storing file corresponding to a protection field.

[0039] The enciphered AOB file and a cryptographic key storing file have the response relation based on the following fixed regulations (1), (2), and (3).

(1) A cryptographic key storing file is arranged at the same directory name as the directory where the enciphered file is stored. The fact that the AOB file is allotted to the SD_Audio directory in the user data area of drawing 9 , and the cryptographic key storing file is also allotted to the SD_Audio directory shows that file arrangement according to this regulation is performed.

[0040] (2) The file name which combined the predetermined extension ".key"

with the head of three characters of the file name of the AOB file in a data area is given to a cryptographic key storing file. When the file name of an AOB file is "AOB001.SA1", it turns out that this head "AOB" of three characters and the file name of "AOBSA1.KEY" which serves as "SA1" from an extension ".key" are given to a cryptographic key storing file as shown in arrow heads nk1 and nk2.

[0041] (3) In the cryptographic key train in a cryptographic key storing file, the serial number which shows the ranking of FileKey which Filekey corresponding to the audio object is located in what position, or corresponds is given to the file name of an AOB file. "File Key Entry#1 in the cryptographic key storing file in drawing 9 , #2, #3 ..... #8" shows the head location of the field where each FileKey in a cryptographic key storing file is stored. On the other hand, serial numbers, such as "001", "002", "003", and "004", are given to the file name of an AOB file. since the serial number in these AOB files means in what position corresponding FileKey is located in a cryptographic key train -- every -- FileKey used when enciphering an AOB file exists in "File Key Entry" which has the same serial number -- things -- ** The arrow heads AK1, AK2, and AK3 in drawing 9 show the response relation between an AOB file and FileKey. That is, it is shown that AOB001.SA1 in a user data area corresponds with FileKey stored in "File Key Entry#1", and AOB002.SA1 supports FileKey stored after "FileKey Entry#2" and FileKey by which AOB003.SA1 was stored after "File Key Entry#3." As shown also in (3) of a more than, FileKey(s) used for encryption of an AOB file differ for every file, and they are stored in "File Key Entry" which has serial numbers, such as "001" included in the file name, "002", "003", and "004", and the same serial number. Since each AOB file is enciphered using different FileKey, even when the cryptographic key of a specific AOB file is exposed, even if other AOB files use exposed FileKey, they cannot cancel encryption temporarily. Damage when FileKey at the time of enciphering an AOB file is exposed by this can be stopped to the minimum.

[0042] {3-3_10-1} The internal configuration of an AOB file, then the internal configuration of an AOB file are explained. Drawing 10 is drawing showing the data configuration of an AOB file hierarchical. The 1st step of this Fig. shows

an AOB file, and the 2nd step shows AOB. The 3rd step shows AOB_BLOCK, the 4th step shows AOB_ELEMENT and the 5th step shows AOB_FRAME.
[0043] "AOB_FRAME" in the 5th step of drawing 10 is a smallest unit which constitutes AOB, and consists of an ADTS header and audio data of an ADTS (Audio Data Transport Stream) format. The audio data of an ADTS format are MPEG 2-AAC. It is stream data which are encoded by [Low Complexity Profile] and reproduced with the transmission speed of 16Kbps - 144Kbps (since it is 1.5Mbps(es), as compared with PCM data, as for the transmission speed of the PCM data recorded on the still more nearly existing compact disk, it turns out that it is lower.). The DS of these AOB_FRAME trains is the same as that of the audio frame train included in the audio data transport distributed in an electronic music distribution. That is, the audio data transport stream which should be stored as an AOB_FRAME train is encoded in MPEG 2-ACC, is in the condition enciphered further, transmits a public network, and is transmitted to a consumer. An AOB file divides the audio data transport stream transmitted such as an AOB_FRAME train, and stores it.
[0044] It is related with the detail of MPEG 2-AAC about {3-3_10-1_11} MPEG 2-AAC, and is ISO/IEC 13818-7:1997 (E). Please refer to Information technology-Generic coding of moving pictures and associated audio information-Part7 Advanced Audio Coding (AAC). here -- being careful -- AOB is a point compressed by the MPEG 2-AAC method which restricted the parameter table described by ISO/IEC 13818-7 like drawing 11 (a), and was applied. Drawing 11 (a) is drawing showing the parameter table described by ISO/IEC 13818-7, and consists of the Parameter column, the Value column, and comment field that show the content of the Comment column.
[0045] The parameter column "profile" shows that the limit of LC-profile specified by ISO/IEC 13838-7 is applied. The parameter column "sampling_frequency#index" shows that sampling frequencies, such as "48kHz, 44.1kHz, 32kHz, 24kHz, 22.05kHZ, 16kHz", are applied. It is shown that the parameter column "number_of_data_block_in_frame" is set as 1header/1 raw_data_block.
[0046] In addition, although AOB_FRAME was explained as what is encoded by the MPEG-AAC method, AOB_FRAME is MPEG-Layer3 (MP3) method

and Windows(trademark) Audio (you may encode by other coding methods, such as a WMA method.). Media Under the present circumstances, the parameter table shown in drawing 11 (b) and drawing 11 (c) must be used instead of the parameter shown in drawing 11 (a).

[0047] {3-3_10-2_12} Although the configuration "AOB_FRAME" of AOB_FRAME contains the audio data encoded under the above limit, the data length of the audio data contained in AOB_FRAME is only data with which the playback time amount serves as 20 mses. However, since an MPEG 2-AAC method is a variable-length-coding method, the data lengths of the audio data contained in each AOB_FRAME differ for every AOB_FRAME. Hereafter, the detail of the configuration of AOB_FRAME is explained, referring to drawing 12 . The 1st step of this Fig. shows a whole AOB_FRAME configuration, and, as for the 2nd step, each part of AOB_FRAME shows how it is enciphered. If this 2nd step is referred to, as for an ADTS header, it turns out that the non-enciphering section, i.e., encryption, is not made. Moreover, audio data include the both sides of the enciphered part and a non-enciphering part. An encryption part arranges two or more 8 bytes of encryption data. 8 bytes of encryption data are generated by enciphering former data of 64 bits using 56-bit FileKey. When encryption is carried out to 64 bitwises such, it leaves a non-enciphering part, without being enciphered in order not to fulfill 64 bits.

[0048] The 3rd step is drawing showing the content of the ADTS header which is a non-enciphering part. An ADTS header is 7 bytes and 12-bit synchronous WORD (set up with FFF), the data length of the audio data contained in the same AOB_FRAME, and the sampling frequency at the time of encoding the audio data are indicated.

{3-3_10-3_13} Cutting tool length setting-out drawing 13 of AOB_FRAME is drawing showing how the cutting tool length of the audio data in each AOB_FRAME is set up in three AOB_FRAME. The data length of audio data #3 by which the data length of audio data #2 by which the data length of audio data #1 contained in AOB_FRAME#1 is contained in x1 and AOB_FRAME#2 in this Fig. is contained in x2 and AOB_FRAME#3 is x3. When each data lengths differ mutually like x1, x2, and x3, to the ADTS header contained in AOB_FRAME#1 A data length x1 is indicated and a data length x3 is

indicated by the ADTS header contained in the ADTS header contained in AOB_FRAME#2 data length x2 and AOB_FRAME#3. Although the audio data itself are enciphered, since the ADTS header itself is not enciphered, if the data length of audio data is read in the ADTS header in each AOB_FRAME, AOB_FRAME which follows can carry out learning of where it exists from. The explanation about AOB_FRAME is finished above.

[0049] {3-3_10-4} AOB_ELEMENT which continues and is located in the 4th step in drawing 10 about AOB_ELEMENT is explained. "AOB_ELEMENT" is the set of two or more continuous AOB_FRAME. Here, AOB_FRAME of the number only of which is contained in AOB_ELEMENT changes according to setting out of sampling_frequency_index shown in drawing 11 (a), and a coding method. That is, the number of AOB_FRAME contained in AOB_ELEMENT is set that the playback time amount of AOB_FRAME contained in the AOB_ELEMENT becomes 2 seconds generally, and turns into a sampling frequency and the different number according to a coding method.

[0050] {3-3_10-5_14} The AOB_FRAME number diagram 14 contained in AOB_ELEMENT is drawing showing a response with sampling_frequency and the AOB_FRAME number contained in AOB_ELEMENT. It will be set to "2", if N shows the playback period of AOB_ELEMENT per second and a coding method is an MPEG-AAC method in this Fig. When sampling_frequency is 48kHz, moreover, the frame number contained in AOB_ELEMENT When it becomes 94 (= 47x2) individual and sampling_frequency is 44.1kHz, The frame number contained in AOB_ELEMENT 86 (= 43x2) individual, When sampling_frequency is 32kHz, the frame number contained in AOB_ELEMENT 64 (= 32x2) individual, When sampling_frequency is 24kHz, a frame number 48 (= 24x2) individual, When sampling_frequency is 22.05kHz, The frame number by which the frame number contained in AOB_ELEMENT is contained in AOB_ELEMENT when 44 (= 22x2) individual and sampling_frequency are 16kHz serves as 32 (= 16x2) individual. However, when division etc. is edited, the AOB_FRAME number of the head of AOB and the last AOB_ELEMENT may become less AOB than the number of drawing 14 .

[0051] Although the information with a special header etc. is not given to AOB_ELEMENT instead, the data length is shown in the time search table. {3-3_10-6_15} Example drawing 15 of the time amount length of AOB_ELEMENT and AOB_FRAME is drawing showing an example of the time amount length of AOB_ELEMENT, and the time amount length of AOB_FRAME. Two or more 1st step of this Fig. is the list of AOB_BLOCK, and the 2nd step shows two or more lists of AOB_ELEMENT. The 3rd step shows two or more lists of AOB_FRAME.

[0052] When this Fig. is referred to, it is equivalent to the playback time amount length of about 2.0 seconds, and, as for AOB_ELEMENT, it turns out that AOB_FRAME in this Fig. deals with playback time amount length called 20msec. the character string "TMSRT_entry" which AOB_ELEMENT is alike, respectively and is attached -- every -- it is shown that the data length of AOB_ELEMENT is indicated by the time search table. With reference to such TMSRT_entry, by performing forward-search playback and hard flow search playback, 2.0 seconds can be skipped and intermission playback of reproducing by 240 mses can be realized.

[0053] {3-3_10-7} Locating [in the 3rd step in drawing finishing the explanation about AOB_ELEMENT and showing the data configuration of the high order of AOB_ELEMENT, i.e., the AOB file of drawing 10 , continuously above about AOB_BLOCK] AOB_BLOCK is explained. "AOB_BLOCK" is a field which consists of effective AOB_ELEMENT, and exists in [one] an AOB file. AOB_BLOCK is equivalent to the playback time amount which made the upper limit playback time amount for 8.4 minutes to AOB_ELEMENT being equivalent to the playback time amount of 2 seconds. every -- the reason which limited AOB to the playback time amount for 8.4 minutes is for controlling the size of a time search table to 504 bytes or less by restricting the number of AOB_ELEMENT contained in AOB_BLOCK.

[0054] {3-3_10-8} Definition of playback time amount explains the reason whose control of a time search table was attained to a detail below control of a time search table. In case playback of forward-search playback and hard flow search playback is performed, "2-second skip 240 ms playback" of skipping 2-second part read-out and reproducing only 240 mses is performed.

Thus, although what is necessary is just to carry out sequential reference of the data length shown in the ADTS header of AOB_FRAME in principle when skipping the time amount length of 2 seconds, in order to skip the time interval of 2 seconds in that case, sequential detection of 100-piece (= 2-second / 20 mses) thing AOB_FRAME will have to be carried out, and an excessive processing load will be given to a regenerative apparatus. In order to mitigate such a processing load, when you describe the read-out place address of the 2-second spacing on a time search table and forward-search playback and hard flow search playback order, just refer to this for a regenerative apparatus. Namely, the data length about each AOB_ELEMENT is described on the time search table, and a regenerative apparatus should just give forward-search playback-hard flow search playback at it to the information for computing the read-out place address 2-second after and 4 second after, and a concrete target with reference to this. It considers how much the data length equivalent to 2 seconds becomes. Since the bit rate at the time of playback of audio data is the range of 16Kbps(es) - 144Kbps as mentioned above, the data length reproduced by per 2 seconds is set to 4 K bytes (=16Kbpsx2/8) - 36 K bytes (=144Kbpsx2/8).

[0055] If the data length per 2 seconds is 4 K bytes - 36 K bytes, 2 bytes (16 bits) of data length of the entry in a time search table to describe the data length of audio data is needed. If 16 bit length is assigned to an entry, it will be because the numeric value of 0 - 64KByte can be described. On the other hand, considering the case where the total data size of a time search table is restricted for example, in 504 bytes (this is data size of TKTMSRT mentioned later), the entry which should be prepared in this time search table must be restricted to 252 (= 504/2) individuals. Since an entry is prepared every 2 seconds as mentioned above, the playback time amount corresponding to 252 entries becomes 504 seconds (=2 second x252), and becomes 8 minutes and 24 seconds (= 8.4 minutes). Thus, data size of a time search table can be made into 504 bytes or less by having restricted the playback time amount in AOB_BLOCK in 8.4 or less minutes.

[0056] {3-3_10-9} AOB is continuously finished the explanation about AOB_BLOCK above and explained about AOB. AOB located in the 2nd step

of drawing 10 is the field where the invalid field was given before and after AOB_BLOCK, and exists in [one] an AOB file. This invalid field is a field which is stored in the same cluster as the AOB_BLOCK concerned, and is written by reading, and AOB_BLOCK and ** concerned. In AOB, it is specified in BIT (explanation about the detail is given in the latter part.) contained in navigation data where [where / from / to] corresponds to AOB_BLOCK.

[0057] Above, it became clear what kind of data are stored in each AOB file. Then, by carrying out reading appearance of AOB and AOB_BLOCK which are contained in eight AOB files shown in drawing 9 continuously explains what kind of content is reproduced.

{3-3_10-10_16} drawing 16 shows what kind of content of playback is reproduced by reproducing continuously each AOB and AOB_BLOCK which are recorded on the AOB file. The 1st step shows eight AOB files in a user data area, and the 2nd step shows eight AOB(s) recorded on each AOB file. The 3rd step shows eight AOB_BLOCK contained in each AOB.

[0058] The 5th step shows the title which consists of the five contents sections. The five contents sections show each of five music called SongA, SongB, SongC, SongD, and SongE, and a title shows the music album which consists of these five music (contents). Broken lines AS1, AS2, and AS3 .... AS7 and AS8 indicate response relation with AOB_BLOCK to be the division part of a music album, and the 4th step shows in what kind of unit the music album of the 5th step is divided.

[0059] if these broken lines are referred to -- every -- the music (SongA) by which AOB_Block contained in AOB#1 is reproduced in the time amount of 6.1 minutes -- it is -- every -- the music (SongB) by which AOB_Block contained in AOB#2 is reproduced in the time amount of 3.3 minutes, and every -- AOB_Block contained in AOB#3 is music (SongC) reproduced in the time amount of 5.5 minutes. It turns out that AOB001.SA1 - AOB003.SA1 are the things corresponding to the music which each became independent of as mentioned above. The 6th step shows the truck sequence which consists of TrackA-E. These TrackA-E supports each of five music called SongA, SongB, SongC, SongD, and SongE, and 1 to 1, and is treated as a playback unit which the piece became independent of.

[0060] On the other hand, AOB#4 are the head part of the music (SongD) reproduced in the time amount of 30.6 minutes, and they are reproduced in the playback time amount of 8.4 minutes. AOB_BLOCK contained in AOB#5 and AOB#6 is the interstitial segment of SongD, and the playback time amount of 8.4 minutes and AOB_BLOCK contained in AOB#7 are parts for the trailer of SongD, and is reproduced in the playback time amount of 5.4 minutes. Thus, as for the music which has the playback time amount of 30.6 minutes, it turns out that it is divided in the unit of (8.4-minute +8.4-minute +8.4-minute + 5.4 minutes), and is contained in each AOB. It turns out that it is stored within the time amount length of [music / that is contained in an AOB file / all] 8.4 minutes in playback time amount length so that he can understand also from this drawing.

[0061] By restricting the playback time amount length of AOB by the above explanation, it became clear that the data size of each time search table matched with AOB is also restricted. Then, the navigation data containing this time search table are explained.

It is as having already stated that {3-3_8A, B-2} navigation data consist of the two files "SD_Audio.PLM" and "SD_Audio.TKM." In a file "SD_Audio.PLM", a file "SD_Audio.TKM" contains a truck manager (TrackManager) including a play list manager (Playlistmanager).

[0062] Although encoded AOB is mentioned in two or more AOB files as explanation of presentation data described, it is not indicated at all which are such playback time amount of AOB and who each AOB is what kind of music name, and a composer is. On the other hand, since two or more AOB(s) are [only being recorded on two or more AOB files and], it is indicated in what kind of sequence no they are reproduced. The truck manager and the play list manager are prepared in order to notify such information to a regenerative apparatus.

[0063] Two or more truck management information which being such playback time amount of AOB and each AOB indicate who it is what kind of music name, and a composer is and many information to be by indicating response relation with a truck to be AOB by which the truck manager is recorded on the AOB file is included here. When a truck is a meaningful

playback unit for a user and tends to store a music work in flash memory card 31, a truck corresponds to music, and if it is a book genre when it is going to store a leading book in flash memory card 31 (a leading book means not books but the document work which read out and was expressed by voice), a truck corresponds to the chapter/knot of a sentence. The truck manager is prepared in order to manage two or more AOB(s) recorded on the AOB file as a set of a truck. [two or more]

[0064] Two or more playback sequence of a truck is specified, and, as for the play list, the play list manager includes two or more such play lists. Henceforth, it explains, referring to a drawing about a truck manager.

{17-1_18} The detail block diagram 17 of Playlistmanager and TrackManager is drawing which detailed gradually the configuration of Playlistmanager in an operation gestalt, and TrackManager, and drawing 18 is drawing showing the size of PlayListManager and TrackManager. That is, the outgoing line which the logical format located in the right column in this Fig. details the logical format located in the left column, and is shown in a broken line clarifies which part within the logical format of the left column the logical format of the right column detailed.

[0065] When the configuration of TrackManager in drawing 17 is referred to according to such a notation, TrackManager is Track Information(it abbreviates to TKI) #1, #2, #3, and #4, as shown in the outgoing line h1 of a broken line..... It consists of #n. These TKI(s) are the information for managing as a truck AOB recorded on the AOB file, and support each AOB file.

[0066] When drawing 17 is referred to, as shown in the outgoing line h2 of a broken line, as for each TKI, it turns out that it consists of Track_Text_Infomation_Data_Area (TKTXTI_DA) text information peculiar to Track_General_Informatin (TKGI) and TKI is described to be, and Track_Time_Serch_Table (TKTMSRT) which has the role of a time search table. When drawing 18 is referred to, the TKI itself is a fixed size (1024 bytes), and it turns out that TKGI and TKTXTI_DA are 512-byte fixed lengths in total. TKTMSRT is also a 512-byte fixed length. Moreover, a maximum of 999 TKI(s) can be set up in TrackManager.

[0067] This TKTMSRT is TMSRT_Header, TMSRT_etry#1, #2, and #3, as

shown in the outgoing line h3 of a broken line..... #n shows becoming.
{17-2_19} Correlation drawing 19 of TKI, and an AOB file and AOB is drawing showing the correlation of TKI shown in drawing 17 , and the AOB file shown in drawing 16 and AOB. The truck sequence which the rectangular-head frame in the 1st step of drawing 19 becomes from TrackA-E, and the rectangular-head frame in the 2nd step of drawing 19 show TrackManager, and the 3rd and the 4th step show eight AOB files shown in drawing 16 . Eight frames in the 5th step show eight AOB(s). Eight AOB(s) shown in drawing 16 are mentioned in these eight AOB files, and they form the music album containing TrackA, TrackB, TrackC, TrackD, and TrackE. The 2nd step shows eight TKI(s). numerical "1" given to these [TKI], "2", "3", and "4" -- every -- the serial number for identifying TKI -- it is -- every -- the serial numbers 001, 002, 003, 004, and 005 with same TKI -- it is matched with the AOB file to which ..... was given. If it is cautious of this point and drawing 19 is referred to, TKI#1 supports AOB001.SA1. It turns out corresponding to [TKI#2 / TKI#4 / SA /1 / AOB004.] corresponding to AOB003.SA1 in AOB002.SA1 and TKI#3 (). [the arrow heads TA1, TA2, and TA3 in this Fig.,] [TA4] ..... every -- it is shown with which AOB file TKI corresponds. . thus, every -- TKI -- every -- since it has AOB recorded on the AOB file, and the response relation of 1 to 1 -- every -- information peculiar to AOB can be indicated in a detail at TKI.
[0068] {17-3_20} As information peculiar to AOB recorded on the AOB file about the DS of TKTMSRT, TKTMSRT is explained first. Drawing 20 is drawing showing the detailed DS of TKTMSRT shown in drawing 17 . On the right-hand side of this Fig., DS with a detailed time search table header (TMSRT_Header) is shown. In drawing 20 , the data size of a time search table header is 8 bytes, and has TMSRT_ID, reserved (from the 2nd byte up to the 3rd byte), and the three fields called Total TMSRT_entry_Number (from the 4th byte up to the 7th byte) (from the 0th byte to the 1st byte). ID which can identify TMSRT uniquely is described by "TMSRT_ID." The total of TMSRT_entry in the TMSRT concerned is described by "Total TMSRT_entry Number."
[0069] {17-3_21-1} It continues about the example of TKTMSRT and TKTMSRT is explained more to a detail. Drawing 21 is drawing showing an

example about TKTMSRT. AOB is shown in the left-hand side of this Fig., and TKTMSRT is shown in it on right-hand side. AOB on the left-hand side of [this] a Fig. is two or more AOB_ELEMENT#1, #2, and #3..... They are two or more fields AR1, AR2, and AR3 which consist of #n and can be set on the right-hand side..... ARn is occupied. Moreover, numeric values, such as "0" in drawing, "32000", "64200", "97000", "1203400", and "1240000", show the relative address to the occupied areas AR1, AR2, and AR3 of each AOB_ELEMENT from the AOB_BLOCK head included in AOB, ARn-1, and ARn. It is shown that AOB_ELEMENT#2 are recorded on the location by which only "32000" was separated from the AOB_BLOCK head. It is shown that AOB_ELEMENT#n -1 is recorded on the location by which only "1203400" was separated from the AOB_BLOCK head by the location from which, as for AOB_ELEMENT#3, only an AOB_BLOCK head to "64200" was separated.

[0070] being careful -- spacing of the start address of each occupied area is not constant value, i.e., every, -- the occupied area of AOB_ELEMENT is that only size different, respectively occupies two or more clusters. The sizes of each occupied area differ, respectively because the sign assignment in each AOB_FRAME is variable length. every -- since the occupancy sizes of AOB_ELEMENT differ -- every -- the case where it jumps at the head of AOB_ELEMENT -- every -- it is necessary to direct beforehand where [in AOB] AOB_ELEMENT exists to a regenerative apparatus It has such an object and two or more TMSRT_entry is indicated. Arrow heads RT1, RT2, and RT3 ..... RTn-1 and RTn are the occupied areas AR1, AR2, and AR3 of each [these] AOB_ELEMENT..... It is ARn-1, ARn, TMSRT_entry#1, TMSRT_entry#2, and TMSRT_entry#3..... The response relation between TMSRT_entry#n -1 and TMSRT_entry#n is shown. That is, it is indicated by TMSRT_entry#1 the size of which the occupied area AR 1 of AOB_ELEMENT#1 occupies, and it is indicated by TMSRT_entry#2 and TMSRT_entry#3 the size of which the occupied areas AR2 and AR3 of AOB_ELEMENT#2 and AOB_ELEMENT#3 occupy.

[0071] Since the occupied area AR 1 occupies from the head of AOB to the head "32000" of AOB_ELEMENT#2, here TMSRT_entry#1 is described to be

32000 (= 32000-0). An occupied area AR 2 Since from the head "32000" of AOB_ELEMENT#1 to the head "64200" of AOB_ELEMENT#2 is occupied TMSRT_entry#2 "32200 (= 64200-32000)", description, and an occupied area AR 3 Since from the head "64200" of AOB_ELEMENT#3 to the head "97000" of AOB_ELEMENT#4 is occupied TMSRT_entry#3 "32800 (= 97000-64200)" occupied-area ARn-1 Since from the head "1203400" of AOB_ELEMENT#n -1 to the head "1240000" of AOB_ELEMENT#n is occupied, TMSRT_entry#n -1 is described to be "36600 (= 1240000-1203400)."

[0072] {17-3_21-2} It turns out that the data size of AOB_ELEMENT is indicated at the read-out method, thus time search table of TKTMSRT. on the other hand, explanation of AOB_ELEMENT described -- as -- every -- as for the data length of AOB_BLOCK, playback time amount becomes within 8.4 minutes -- as -- laws -- ****** -- it is that and the total of AOB_ELEMENT contained in one AOB is stopped below at the predetermined number (252 pieces shown in drawing 20). Since an AOB_ELEMENT number is stopped below at a predetermined number, the total of TMSRT_entry corresponding to AOB_ELEMENT also becomes below a predetermined number, and the data size of TKTMSRT containing these also becomes below predetermined size. Since the size of TKTMSRT was controlled, a regenerative apparatus can read and use TKI as follows.

[0073] If reading appearance of a certain AOB is carried out and the playback is started, TKI corresponding to it is read and it stores in memory. Henceforth, this TKI is stored in memory in the period which the playback concerned of AOB is continuing. If the playback concerned of AOB finishes, reading appearance of the AOB which follows this will be carried out and the playback will be started, TKI corresponding to it will be read and TKI stored on memory till then will be overwritten using TKI by which reading appearance was newly carried out. Henceforth, this TKI is stored in memory in the period which the playback concerned of AOB is continuing.

[0074] If read-out of TKI and storing in memory are performed in this way, even if the amount of mounting of the memory in a regenerative apparatus is small-scale, special playbacks, such as forward-search playback and hard flow search playback, can be performed only by reading required TKI. In

addition, with this operation gestalt, although the data length from the start address of a certain AOB_ELEMENT to the start address of following AOB_ELEMENT was indicated as TMSRT_entry, the relative address from the head of AOB_BLOCK to the head of each AOB_ELEMENT may be indicated.

[0075] {17-3_21-3} It explains how AOB_ELEMENT of arbitration should be read to the specific last of the cluster containing AOB_ELEMENT with reference to TKTMSRT. What is necessary is to ask for the cluster u which fills the following {formulas 1}, and just to read the offset v or subsequent ones from the head of the cluster u, when reading AOB_ELEMENT#y located in the y-th from a head in AOB with reference to TKTMSRT the size of each AOB_ELEMENT was indicated to be.

{Formula 1}

cluster u =/(total +DATA_Offset of TMSRT_entry from AOB_ELEMENT#1 to AOB_ELEMENT#y -1)cluster size offset v =(total +DATA_Offset of TMSRT_entry from AOB_ELEMENT#1 to AOB_ELEMENT#y -1) mod cluster size c =a mod b -- in a certain case, c shows the remainder at the time of breaking a by b, and DATA_Offset is information indicated by BIT and is mentioned later.

[0076] {17-4} Above, explanation of a time search map (TKTMSRT) is finished about TKTXTI_DA. Next, Track Text Information Data Area (TKTXTI_DA) indicated in drawing 17 on the upper case of TKTMSRT is explained. The text information which shows an artist name, an album name, an arrangement person name, a producer name, etc. is described by Track Text Information Data Area (TKTXTI_DA). This field is secured even when text data does not exist.

[0077] {17-5} TKGI which continues about TKGI and is in the upper case of TKTXTI_DA is explained. In drawing 17 , as shown in the outgoing line h4 of a broken line, TKGI of TKI The identifier "TKI_ID" of TKI, a TKI number "TKIN", the size of TKI "TKI_SZ", The link pointer "TKI_LNK_PTR" to the next TKI, a block attribute "TKI_BLK_ATR", It turns out that a series of information of playback time amount "TKI_PB_TM", the audio attribute "TKI_AOB_ATR" of TKI, "ISRC", and block information "BIT" is recorded (in addition, this Fig. for

simplification of explanation). it has omitted and written about some fields. .
[0078] {17-5_22-1} The detail configuration of TKGI is explained referring to drawing 22 hereafter about TKGI. The bit pattern of "TKI_BLK_ATR", "TKI_AOB_ATR", and "ISRC" by which the data configuration of TKGI shown in drawing 17 is arranged on the left-hand side in drawing, and the difference between this Fig. and drawing 17 was not clarified by drawing 17 is the point arranged the right-hand side in drawing.
[0079] {17-5_22-2} ID (the code in this operation gestalt 2 bytes of "A4") which can identify TKI uniquely is described by "TKI_ID" about TKI_ID.
{17-5_22-3} The TKI number of the range from one to 999 is described by "TKIN" about TKIN. In addition, this TKI number must not overlap the TKI number described by TKIN of other TKI(s). It shall describe in what position TKI is located in the ranking of TKI in TrackManager, i.e., TrackManager, as such TKIN. If it is TKI#1 in this Fig., a TKI number is indicated to be "1", and a TKI number is indicated to be "3", if it is TKI#2 and TKI numbers are "2" and TKI#3.
[0080] {17-5_22-4} The data size of TKI is described by "TKI_SZ" per byte count about TKI_SZ. By drawing 22 , since the data size of TKI is specified as 1024 bytes, this operation gestalt is described to be 1024 bytes.
{17-5_22-5} TKIN about TKI of the link place of TKI concerned is described by "TKI_LNK_PTR" about TKI_LNK_PTR. Here, the response relation between TKI(s) is explained.
[0081] When a truck consists of two or more AOB(s) and they are recorded on two or more AOB files, two or more TKI(s) matched with the AOB file of these plurality will be united, and will manage the truck concerned. Thus, when two or more TKI(s) are united, it needs to be shown the AOB file corresponding to which TKI follows the AOB file corresponding to these [TKI]. TKI_LNK_PTR is used for the application of describing TKIN about each TKI which follows TKI.
[0082] {17-5_22-6_19} It explains how TKI_LNK_PTR is set up after that in eight TKI(s) shown in drawing 19 about TKI_LNK_PTR. In TKI#1-TKI#3 which constitute one truck, and TKI#8, although the TKI_LNK_PTR is not set up, TKI#4 corresponding to four AOB files which constitute TrackD, TKI#5, TKI#6,

and TKI#7 are set up so that each TKI_LNK_PTR may direct following TKI_LNK_PTR. That is, as shown in arrow heads tangent line4, tangent line5, and tangent line6, TKI_LNK_PTR of TKI#4 is directing TKI#5, TKI_LNK_PTR of TKI#5 directs TKI#6 and TKI_LNK_PTR of TKI#6 is directing TKI#7. Each of these constitutes TrackD. By referring to these TKI_LNK_PTR in TKI matched with four AOB files shows that four AOB files called four TKI(s) called TKI#4-TKI#7 and AOB004.SA1 - AOB007.SA1 constitute TrackD in one.

[0083] {17-5_22-7} The attribute about TKI is described by "TKI_BLK_ATR" about TKI_BLK_ATR. The bit pattern of TKI_BLK_ATR is shown in the frame pulled out with the broken line from TKI_BLK_ATR in drawing 22 . In this Fig., TKI_BLK_ATR is 16 bits and even b15 bit is secured from b triplet for the future extension. The attribute about TKI is described using the triplet from the bit number b2 to b0.

[0084] TKI is used, and when one TKI is contained on one truck, the value of "000b" is described by TKI_BLK_ATR (this setting out is henceforth called "Track".). TKI is used, and including TKI of plurality [truck / one], when the TKI concerned is that head, the value of "001b" is described by TKI_BLK_ATR (this setting out is henceforth called "Head_of_Track".). TKI is used, one truck consists of two or more TKI(s), and when the TKI concerned is that medium, the value of "010b" is described by TKI_BLK_ATR (this setting out is henceforth called "Midpoint_of_Track"). TKI is used, one truck consists of two or more TKI(s), and when the TKI concerned is that termination, the value of "011b" is described by TKI_BLK_ATR (this setting out is henceforth called "End_of_Track".). When it is TKI which whose TKI is intact and has the field of TKI and which was case [TKI] namely, deleted, the value of "100b" is described (this setting out is henceforth called "Unused"). TKI is intact, and when there is no field of TKI (i.e., when it is TKI of an initial state), the value of "101b" is described.

[0085] {17-5_22-8_19} An example of example drawing 19 of setting out of TKI_BLK_ATR explains how each TKI_BLK_ATR about TKI is set up. If each TKI_BLK_ATR in TKI is referred to, TKI#1 (AOB001.SA1), four called TKI#2 (AOB002.SA1), TKI#3 (AOB003.SA1), and TKI#8 (AOB008.SA1) --

constructing -- Since the truck with which each became independent is supported, TKI_BLK_ATR of TKI#1, TKI#2, TKI#3, and TKI#8 is set up with "Track."

[0086] As for TKI_BLK_ATR in TKI#7, it turns out that TKI_BLK_ATR in TKI#4 is set up with "Head_of_Track" and "End_of_Track", TKI#5, and TKI#6 are set up with "Midpoint_of_Track." As for TKI#4 (AOB004.SA1) in which this has TKI#4 and response relation, the head section of a truck, TKI#5 and TKI#5 in which it has TKI#6 and response relation (AOB005.SA1), and TKI#6 (AOB006.SA1) mean that the pars intermedia of a truck, and TKI#7 and TKI#7 which has response relation (AOB007.SA1) are the trailers of a truck.

[0087] thus, every -- if **** of TKI (AOB file) is classified according to the publication of TKI_BLK_ATR in TKI, it turns out that TKI#1 (AOB001.SA1) constitutes the 1st truck (TrackA). It turns out that TKI#2 (AOB002.SA1) constitutes the 2nd truck (TrackB), and TKI#3 (AOB003.SA1) constitutes the 3rd truck (TrackC). It turns out that TKI#4 (AOB004.SA1) constitutes the head part of the 4th truck (TrackD), TKI#5 (AOB005.SA1) and TKI#6 (AOB006.SA1) constitute the interstitial segment of TrackD, and TKI#7 (AOB007.SA1) constitutes a part for the trailer of TrackD. It turns out that TKI#8 (AOB008.SA1) constitutes a part for the trailer of the 5th TrackE independently.

[0088] {17-5_22-9} The playback time amount of the truck (music) constituted by AOB recorded on the AOB file corresponding to TKI is described by "TKI_PB_TM" about TKI_PB_TM. When a truck consists of two or more TKI(s), the playback time amount of the whole truck is described by TKI_PB_TM about top TKI. Moreover, the playback time amount of AOB corresponding to each TKI is described by TKI of the 2nd henceforth.

[0089] {17-5_22-10} The encoding conditions at the time of generating AOB(s), such as with what kind of sampling frequency AOB recorded on the AOB file corresponding to TKI is sampled by "TKI_AOB_ATR" about TKI_AOB_ATR, with what kind of bit rate it is transmitted, or which is the number of channels, are described. The frame pulled out with the broken line from "TKI_AOB_ATR" shows the bit pattern of TKI_AOB_ATR. In this Fig., TKI_AOB_ATR is 20 bits and coding mode is described by the field from the bit number b16 to the bit

number b19. When are encoded by MPEG-2 AAC (with ADTS header), the value of "0000b" is encoded by MPEG-layer3 (MP3) and the value of "0001b" is encoded by Windows Media Audio (WMA), "0010b" is described, respectively.

[0090] A bit rate is described by the field from the bit number b15 to the bit number b8. When encoded by MPEG-2 AAC (with ADTS header) When the value of "16"-" 72" is encoded by MPEG1-layer3 (MP3), the value of "16"-" 96" When are encoded by MPEG 2-layer3 (MP3) LSF and the value of "16"-" 80" is encoded by Windows Media Audio (WMA), the value of "8"-" 16" is described, respectively.

[0091] A sampling frequency is described from the bit number b7 by the bit number b4. As for the case of "0010b" and 24kHz, in the case of 48kHz, in the case of "0000b "case of 44.1kHz" 0001b", and 32kHz, the value of "0101b" is described in the case of "0100b" and 16kHz, as for the case of "0011b" and 22.05kHz. The number of channels is described by the field from the bit number b3 to the bit number b1. In 1ch (mono), "000b" is described. In 2ch(es) (stereo), "001b" is described.

[0092] The field of the bit number b31 to the bit number b20 and the bit number b0 is reserved for future extensions.

{17-5_22-11} ISRC (International Standard Recording Code) in TKGI is described by "ISRC" about ISRC. The frame pulled out with the broken line from "ISRC" in drawing 22 shows the content of ISRC. ISRC consists of 10 bytes, Recording-item code (#12) is described by the field from the bit number b4 to the bit number b7, and Recording code/Recording-item code (#11) is described by the field from the bit number b8 to the bit number b11 as shown in this frame.

[0093] Recording code (ISRC#10, #9, #8) is described by the field from the bit number b12 to the bit number b23. Year-of-Recording code (ISRC#6, #7) is described by the field from the bit number b24 to the bit number b31. Henceforth, First Owner Code (ISRC#3, #4, #5) is described by the field from the bit number b32 to the bit number b37, the field from the bit number b40 to the bit number b45, and the field from the bit number b48 to the bit number b53. Country code (ISRC#1, #2, #3) is described by the field from the bit

number b56 to the bit number b61, and the field from the bit number b64 to the bit number b69. 1-bit Validity flag is described by the field of the bit number b79. In addition, about the detail of ISRC, it is ISO3901. : Please refer to 1986"Documentation-International Standard Recording Code (ISRC)".

[0094] {17-5_22-12_23A-1} It is the table on which "a block information table (BIT)" manages AOB_BLOCK about BIT. Drawing 23 (a) and (b) are drawings showing the detail configuration of BIT. The DATA_OFFSET field where BIT occupies even the 63rd byte from the 60th byte as shown in drawing 23 (a), The SZ_DATA field which occupies even the 67th byte from the 64th byte, The TMSRTE_Ns field which occupies even the 71st byte from the 68th byte, The FNs_1 st_TMSRTE field which occupies even the 73rd byte from the 72nd byte, The FNs_Last_TMSRTE field which occupies even the 75th byte from the 74th byte, It consists of the FNs_Middle_TMSRTE field which occupies even the 77th byte from the 76th byte, and the TIME_LENGTH field which occupies even the 79th byte from the 78th byte. Hereafter, explanation of each component is given.

[0095] {17-5_22-12_23A-2} The relative address from a cluster boundary to the head of each AOB_BLOCK is described by "DATA_OFFSET" per cutting tool about DATA_Offset. Thereby, it is expressed whether only in which, an invalid field exists in from AOB before AOB_BLOCK. The music stored in flash memory card 31 as AOB is the music recorded by carrying out an air check, and when a D.J.'s voice is mixed with the part of the intro of that music, this unnecessary voice is excepted from AOB_BLOCK and it can avoid reproducing it by setting up DATA_Offset in BIT.

[0096] {17-5_22-12_23A-3} The data length of each AOB_BLOCK is described by "SZ_DATA" per cutting tool about SZ_DATA. If the value adding SZ_DATA and DATA_Offset is deducted from the file size (integral multiple of cluster size) in which AOB is mentioned, it can ask for the size of which the invalid field which follows AOB_BLOCK is.

[0097] {17-5_22-12_23A-4} The total of TMSRT_entry contained in each AOB_BLOCK is described by "TMSRTE_Ns" about TMSRTE_Ns.

{17-5_22-12_23A-5} The AOB_FRAME number contained in AOB_ELEMENT located in the head in the AOB_BLOCK concerned is described by "FNs_1

st_TMSRTE" about "FNs_1 st_TMSRTE", "FNs_Last_TMSRTE", and "FNs_Middle_TMSRTE."

[0098] The number of AOB_FRAME contained in AOB_ELEMENT at the tail end of AOB_BLOCK is described by "FNs_Last_TMSRTE." The number of AOB_FRAME contained in AOB_ELEMENT except AOB_ELEMENT at a head and the tail end, i.e., AOB_ELEMENT located in the pars intermedia of AOB_BLOCK, is described by "FNs_Middle_TMSRTE."

[0099] "TIME_LENGTH" is the field which describes the playback period of AOB_ELEMENT in the time amount precision of ms order by the format shown in drawing 23 (c). If the TIME_LENGTH field is 16 bit length and coding methods are a MPEG-AAC method and MPEG-Layer3 method, as shown in drawing 23 (c), since the playback period of AOB_ELEMENT will become 2 seconds, the value of 2000 is described by TIME_LENGTH.

[0100] {17-5_22-13_23B} drawing 23 (b) is drawing showing how many AOB_FRAME is stored in FNs_Middle_TMSRTE. This Fig. shows sampling_frequency and an AOB_FRAME number of response relation which are contained in AOB_ELEMENT of pars intermedia like drawing 14 . The response relation between sampling_frequency in this Fig. and the frame number contained in AOB_ELEMENT is completely the same as that of drawing 14 , and it turns out that it is the different number according to a sampling frequency. Although the frame number in "FNs_1 st_TMSRTE" and "FNs_Last_TMSRTE" is set as the frame number in "FNs_Middle_TMSRTE", and the frame number of principle identitas, when setting an invalid field as AOB_ELEMENT located in the head or tail of AOB_BLOCK, "FNs_1 st_TMSRTE" and "FNs_Last_TMSRTE" serve as a different value from "FNs_Middle_TMSRTE."

[0101] {17-5_22-14_24} Example drawing 24 of storing of AOB_ELEMENT is drawing showing the cluster 007 in which AOB which consists of AOB_ELEMENT#1-#4 is stored - cluster 00E. As AOB shows drawing 24 , when it is stored, it explains how BIT is set up. Although the triangular pennant-like notation is given to each of AOB_ELEMENT#1-AOB_ELEMENT#4 stored in these clusters 007 - cluster 00E, these show that TMSRT_entry contained in TKI is set as each of AOB_ELEMENT#1-

AOB_ELEMENT#4.

[0102] Under the present circumstances, a part for the point of AOB_ELEMENT#1 in an AOB head is stored in the cluster 007, and a part for the trailer of AOB_ELEMENT#4 in an AOB tail is stored in cluster 00E. AOB_ELEMENT#1-#4 occupy even md4 in the middle of cluster md0 to 00E in the middle of a cluster 007. SZ_DATA in BIT is directing even the last of AOB_ELEMENT#1 to AOB_ELEMENT#4, as shown in an arrow head sd1, is a field in a cluster 007 and 00E, and is not directing the parts ud0 and ud1 which are not occupied by AOB_ELEMENT.

[0103] On the other hand, AOB is a field in a cluster 007 and cluster 00E, and is included to the parts ud0 and ud1 which are not occupied by AOB_ELEMENT#1 and AOB_ELEMENT#4. DATA_Offset in BIT is directing the relative value from the data length of the non-occupying part ud0, i.e., the head of a cluster 007, to the head of AOB_ELEMENT#1. In this Fig., AOB_ELEMENT#1 occupies even md1 from md0 in the middle of a cluster 008 in the middle of a cluster 007. This AOB_ELEMENT#1 does not occupy the cluster 008 whole and it is occupied by AOB_ELEMENT#2 after that trailer part. AOB_ELEMENT#4 occupy even the part md4 from the part md3 in the middle of cluster 00E in the middle of cluster 00C. Thus, it turns out at AOB_ELEMENT that what is recorded exists so that the boundary of a cluster may be straddled. That is, the boundary of a cluster is [AOB_ELEMENT] completely unrelated and it is recorded. "FNs_1 st_TMSRTE" in BIT shows the frame number of AOB_ELEMENT#1 in a cluster 007 - a cluster 008, and "FNs_Last_TMSRTE" in BIT shows the frame number of AOB_ELEMENT#4 in cluster 00C - cluster 00E.

[0104] Thus, as for each AOB_ELEMENT, it turns out that it is arranged freely not related on the boundary of a cluster, and the offset from a cluster boundary to AOB_ELEMENT and the frame number for every AOB_ELEMENT are managed by BIT.

{17-5_22-14_25} It explains below how the frame number for every AOB_ELEMENT indicated by directions 1BIT of the frame number for every AOB_ELEMENT is used. The frame number indicated by BIT skips playback progress time of day to the 1st for 2 seconds first, and when performing

forward-search playback of reproducing only 240 mses, and hard flow search playback, it is used.

[0105] Drawing 25 is drawing showing how AOB_FRAME#x +1 which should be reproduced next is set up, when performing forward-search playback from AOB_FRAME#x in AOB_ELEMENT#y of the arbitration in AOB. This Fig. is drawing plotted supposing the case where forward-search playback is directed, when AOB_FRAME#x contained in AOB_ELEMENT#y is reproduced. In this Fig., the frame number and intermittent skip time amount skip_time by which t is equivalent to predetermined intermittent playback time amount (=240 ms), and f (t) is equivalent to intermittent playback time amount set to f (skip_time) the time amount length (in this case, 2 seconds) which should skip in case intermittent playback is performed, and the frame number corresponding to this intermittent skip time amount skip_time. Intermittent playback is performed by repeating the procedure of following ****** here.

[0106] ** Jump at the head of a flag (AOB_ELEMENT) with reference to TMSRT_entry indicated by TKTMSRT.

** ** to which only 240 mses are reproduced -- jump at the head of the following flag (AOB_ELEMENT).

In addition, this operation gestalt explains how to realize more exact intermittent playback of carrying out 240 ms playback, jumping in the part of 2 seconds after, and carrying out 240 ms playback.

[0107] From AOB_FRAME#x contained in AOB_ELEMENT#y, AOB_FRAME#x +1 after 2 second +240 ms should exist in AOB_ELEMENT#y +1. Although the start address about following AOB_ELEMENT#y +1 can be immediately computed by reading TMSRT_entry in TKTMSRT when it specifies AOB_FRAME#x +1 after 2 second +240 ms, the AOB_FRAME number which intervenes by AOB_FRAME#x +1 from the start address of the AOB_ELEMENT#y +1 cannot be known only by TMSRT_entry. In order to compute such an AOB_FRAME number, it is necessary to ask by deducting the total frame number contained in AOB_ELEMENT#y from the sum of #x which show in what position AOB_FRAME#x is located from the head of AOB_ELEMENT#y, and f (t) and f (skip_time). Such, in order to compute simply the relative frame

location of AOB_FRAME#x +1 in following AOB_ELEMENT#y +1, "FNs_1 st_TMSRTE" about each AOB_ELEMENT, "FNs_Middle_TMSRTE", and "FNs_Last_TMSRTE" are indicated to BIT.

[0108] {17-5_22-15_26A} The frame number indicated by directions 2BIT of the frame number for every AOB_ELEMENT is used in case the function (time search function) 2nd to start playback from the playback time of day of arbitration is performed. Drawing 26 (a) is drawing showing how AOB_ELEMENT corresponding to the appointed time of day and AOB_FRAME are specified, when the playback start time of arbitration is specified.

[0109] What is necessary is just to start the AOB_FRAME location x to AOB_ELEMENT#y which fills the following formulas, and playback, if the playback appointed time of day is made into Jmp_Entry (second) when the time of day of arbitration is specified and playback is directed.

{Formula 2}

Since Jmp_Entry(second) =(FNs_1 st_TMSRTE+FNs_middle_TMSRTExy+x) x20msec these "FNs_1 st_TMSRTE" and "FNs_Middle_TMSRTE" are indicated by BIT If AOB_ELEMENT#y and AOB_FRAME#x are computed by applying these to {a formula 2} It asks for the start address of AOB_ELEMENT#y +2 located in the y+2nd in AOB with reference to TKTMSRT corresponding to this AOB. If retrieval of AOB_FRAME#x is begun and it is searched for x-th AOB_FRAME from this start address, playback will be started from this x-th AOB_FRAME. Thereby, playback can be started from the time of day specified in Jmp_Entry (second).

[0110] Under the present circumstances, since what is necessary is not to search a part for the ADTS header unit of an AOB file, but just to refer to the AOB_ELEMENT unit TMSRT_entry is described to be by TKTMSRT, the playback location corresponding to the playback appointed time of day is discoverable at a high speed. What is necessary is just to compute AOB_ELEMENT#y which fills the following {formulas 3}, and AOB_FRAME#x, when similarly a time search function is performed and Jmp_Entry (second) is specified to the truck which consists of two or more AOB(s).

{Formula 3}

Jmp_Entry (second) total +(FNs_1 st_TMSRTE(#n+1)+FNs_middle_TMSRTE (#n+1) and y+x) - of the playback time amount from = AOB#1 to AOB#n -- the total of the playback time amount of AOB from AOB#1 to AOB#n is as follows 20 msec here.

Total [of the playback time amount from AOB#1 to AOB#n] = () ["FNs_1 st_TMSRTE"] (#1) + "FNs_Middle_TMSRTE" - (#1) (TMSRT_entry number #1-2)+ "FNs_Last_TMSRTE" (#1) + "FNs_1 st_TMSRTE" (#2)+"FNs_Middle_TMSRTE" (#2) and TMSRT_entry number #2-2+ "FNs_Last_TMSRTE" (#2) + "FNs_1 st_TMSRTE" (#3)+"FNs_Middle_TMSRTE" (#3) and TMSRT_entry number #3-2+ "FNs_Last_TMSRTE" (#3) ..... + "FNs_1 st_TMSRTE" (#n)+ "FNs_Middle_TMSRTE" (#n) AOB#n which fills -TMSRT_entry number #n-2+"FNs_Last_TMSRTE" (#n) and 20msec {a formula 3}, If AOB_ELEMENT#y and AOB_FRAME#x are computed If retrieval of AOB_FRAME#x is begun and it is searched for x-th AOB_FRAME from the address located in y+2nd AOB_ELEMENT#y +2 with reference to TKTMSRT corresponding to this AOB#n +1 Playback is started from this x-th AOB_FRAME.

[0111] {17-5_22-16_27A, B} In the place explaining all the information included in the deletion TKI of an AOB file and TKI When some trucks were deleted (case1), after some trucks were deleted, [when dividing one truck when recording a new truck (case2) and unifying two of two or more trucks of arbitration on one truck (case3), and obtaining two trucks (case4)] It explains how TKI is updated.

[0112] The case (case1) where some trucks are deleted is explained first. Drawing 27 (a) and (b) are drawings supposing the case where a truck is deleted. The operator shall wish for this Fig. to show TrackManager shown in drawing 19 , and to delete TrackB in this Fig. Since AOB corresponding to this TrackB is recorded on AOB002.SA1 and it is matched with TKI#2, while AOB002.SA1 is deleted, TKI_BLK_ATR of TKI#2 is set as "Unused." AOB002.SA1 is deleted and the condition that TKI_BLK_ATR of TKI#2 was set as "Unused" is shown in drawing 27 (b). Since AOB002.SA1 was deleted, the field which AOB002.SA1 occupied in the data area is released by the free area. It turns out that TKI_BLK_ATR of TKI#2 is set as "Unused" in

TrackManager with it.

[0113] {17-5_22-17_28A, B} After assignment of TKI in the case of newly recording an AOB file, then some trucks are deleted, the case (case2) where a new truck is recorded is explained. Drawing 28 (a) is drawing showing TrackManager after deletion of a truck was performed two or more times. In this Fig., if two or more trucks are deleted and these are matched with TKI#2, TKI#4, TKI#5, TKI#7, and TKI#8, these TKI_BLK_ATR of TKI is set as "Unused." Although deletion of an AOB file is performed like the usual file, TKI_BLK_ATR of TKI to which TrackManager corresponds completes [only being set as "Unused", and] deletion. When it does so, as shown in this Fig., TKI of "Unused" will appear on TrackManager in the shape of vermin.

[0114] Drawing 28 (b) is drawing showing how the writing is performed, when TKI of "Unused" exists and it writes in new TKI and an AOB file here. The case where it is going to write in the truck which consists of four AOB(s) here is assumed. DPL_TK_SRP which mentions later which opening TKI is assigned to record of AOB here is determined, or TKI of arbitration is assigned. In TrackManager, TKI#2 set as "Unused", TKI#4, TKI#7, and TKI#8 are assigned to the four AOB(s).

[0115] Since these four AOB(s) constitute one truck, "Midpoint_of_Track" and TKI_BLK_ATR about TKI#8 are set [TKI_BLK_ATR / about TKI#2] up with "End_of_Track" in "Head_of_Track" and TKI_BLK_ATR about TKI#4 and TKI#7. Four TKI#2 which constitute Truck TrackD, TKI#4, TKI#7, and TKI#8 are set up so that each TKI_LNK_PTR may direct following TKI_LNK_PTR which constitutes Truck TrackD. That is, as shown in arrow heads tangent line2, tangent line4, and tangent line7, TKI_LNK_PTR of TKI#2 is directing TKI#4, TKI_LNK_PTR of TKI#4 directs TKI#7 and TKI_LNK_PTR of TKI#7 is directing TKI#8.

[0116] Then, TKI#2, TKI#4, TKI#7, four file AOB002.SAs1 that have the same number as each of TKI#8, AOB004.SA1, AOB007.SA1, and AOB008.SA1 are created, and four AOB(s) which constitute TrackD in these four files are recorded. the 4th truck TrackD is managed by setting out of this TKI_BLK_ATR and TKI_LNK_PTR using TKI#2, TKI#4, TKI#7, and TKI#8 -- things -- **

[0117] As mentioned above, when newly writing a truck in flash memory card 31, it turns out that TKI set as TrackManager by "Unused" till then is assigned to TKI about the truck which should be recorded newly.

{17-5_22-18_29A, B} The renewal of TKI at the time of performing integration (case3) of TKI setting out in the case of unifying two trucks, then a truck is explained.

[0118] Drawing 29 (a) and (b) are drawings showing how TKI is set up, when unifying two trucks to one. Drawing 29 (a) shall be the same as that of TrackManager shown in drawing 19 , and the operator shall wish the editing operation of unifying TrackC and TrackE on one truck, in drawing 29 (a). Since AOB corresponding to these TrackC(s) and TrackE is recorded on AOB003.SA1 and AOB008.SA1 and they are matched with TKI#3 and TKI#8, rewriting of TKI_BLK_ATR of these TKI#3 and TKI#8 is performed. Drawing 29 (b) is drawing showing the rewriting back of TKI_BLK_ATR of TKI. Although TKI_BLK_ATR of TKI#3 and TKI#8 is indicated to be Track in this Fig., in drawing 29 (b), TKI_BLK_ATR of TKI#3 is rewritten by "Head_of_Track" and TKI_BLK_ATR of TKI#8 is rewritten by "End_of_Track." Thus, TKI#3, TKI#8, AOB003.SA1 corresponding to these, and AOB008.SA1 are treated as one truck called TrackC by rewriting TKI_BLK_ATR. In addition, it is rewritten so that TKI_LNK_PTR of TKI#3 may direct TKI#8 as a link place.

[0119] here -- it should mind -- although TKI_BLK_ATR of TKI was rewritten, processing in which AOB003.SA1 and AOB008.SA1 are unified is the point which was not performed. It is because these AOB files need to decode the AOB file enciphered when unifying these to one, since it was enciphered in mutually different FileKey, two processings of decryption-encryption of enciphering again need to perform them about each AOB file and a great quantity of processing loads are required. Moreover, since the AOB file after integration is enciphered in one FileKey, as compared with integration before, it is because weakened protection of copyrights is caused.

[0120] In addition, TKI is because there is a possibility that the size of TKI after integration may become large too much, when unifying this in editing operation one, although the size of TKTMSRT is set not to become large from the first. As mentioned above, while integration edit of the truck in this

operation gestalt had maintained encryption of an AOB file, it turns out that it realizes only by attribute modification of TKI_BLK_ATR.

[0121] {17-5_22-18_29A, B-1_30, 31} Although it is as having mentioned above that integration of the condition truck which should be filled when unifying a truck is realized by attribute modification of TKI_BLK_ATR, it is required that AOB contained on the truck integrated should fulfill the following conditions in integration of a truck. The 1st condition is that the audio attribute (audio coding mode, a bit rate, a sampling frequency, the number of channels) of AOB contained on the truck which follows, and AOB contained on the truck to precede is in agreement. If these differs in the audio attribute of AOB by AOB of order, a regenerative apparatus once needs to reset actuation of a decoder and will depend it on the reason it becomes difficult to reproduce two continuous AOB(s) seamlessly (without breaking off).

[0122] The 2nd condition is that three or more AOB(s) which an AOB_FRAME number becomes only from AOB_ELEMENT with which "FNs_Middle_TMSRTE" is not filled do not continue in the truck obtained by the integration back. AOB is classified into two types according to whether at least one of AOB_ELEMENT has AOB_FRAME of the frame number directed by "FNs_Middle_TMSRTE", and the same number. AOB of the 1st type is AOB which has at least one AOB_ELEMENT which has AOB_FRAME of the frame number directed by "FNs_Middle_TMSRTE", and the same number, and AOB of the 2nd type is AOB which does not have any AOB_ELEMENT which has AOB_FRAME of the frame number directed by "FNs_Middle_TMSRTE", and the same number. That is, AOB_ELEMENT in AOB of the 2nd type is less than the frame number all were instructed to be by "FNs_Middle_TMSRTE", and the 2nd condition mentioned above forbids that three or more AOB(s) of Type2 continue. The reason for prohibition is as follows. When AOB of Type2 is continuing, it becomes impossible that is, to fill the buffer in a regenerative apparatus with AOB_FRAME, although being filled with a sufficient number of AOB_FRAME is desirable as for the buffer in a regenerative apparatus in case AOB is read one by one. When it does so, it becomes impossible to maintain the continuity of a lifting and playback of AOB of an underflow of the buffer in a regenerative apparatus. In order to

avoid generating of such an underflow, the 2nd condition forbids that three or more AOB(s) of Type2 continue.

[0123] Drawing 30 (a) shows AOB of Type1, and drawing 30 (b) is drawing showing AOB of Type2. AOB in drawing 30 (b) consists only of two or less AOB_ELEMENT, and these two or less AOB_ELEMENT does not have AOB_FRAME shown in "FNs_Middle_TMSRTE" (in addition, only FNs_1 st_TMSRTE is described by BIT in this case.). it is classified into AOB of this Type2 even if it is AOB constituted by only one AOB_FRAME, since it is the requirements for Type2AOB not to have AOB_FRAME shown in "FNs_Middle_TMSRTE" -- things -- **

[0124] Drawing 31 (a) is the combination of Type1+Type2+Type2+Type1, and is drawing showing the case where a multiple track is unified to one. In this case, since it is avoided that three AOB(s) of Type2 continue, these are unified by one truck. Drawing 31 (b) is the combination of Type1+Type2+Type2+Type2+Type1, and is drawing showing the case where a multiple track is unified to one. In this case, since three AOB(s) of Type2 are continuing, unifying these on one truck is forbidden.

[0125] {17-5_22-18_29A, B-1_32} This truck can be unified with the truck which allotted AOB of Type2 to the head, or the truck which allotted AOB of Type1 to the head when the termination of the truck to precede is Type1 according to integration of the truck shown in truck integrated drawing 31 (a) in consideration of the combination of Type1 and Type2. Drawing 32 (a) is drawing showing the arrangement pattern with which AOB of Type1 is allotted to the termination of the truck to precede, and AOB of Type1 is allotted to the head of the truck which follows. Moreover, drawing 32 (b) is drawing showing the arrangement pattern with which AOB of Type1 is allotted to the termination of the truck to precede, and AOB of Type2 is allotted to the head of the truck which follows. Since each of these fulfills conditions 2, they can be unified on one truck.

[0126] The termination of the truck to precede is Type2, when AOB of Type1 is arranged just before that Type2, AOB of Type2 is allotted to the truck of Type1, or a head, and a head can unify this truck with the truck with which AOB of Type1 has been arranged just behind that. Drawing 32 (c) is drawing

showing the arrangement pattern with which AOB is allotted to the termination of the truck to precede in Type1 and Type2 order, and AOB of Type1 is allotted to the head of the truck which follows. Drawing 32 (d) is drawing showing the arrangement pattern with which AOB is allotted to the termination of the truck to precede in Type1 and Type2 order, and AOB of Type2 and Type1 is allotted to the head of the truck which follows. Since these also fulfill conditions 2, they can be unified on one truck.

[0127] The termination of the truck to precede is Type2, and this truck can be unified with the truck with which AOB of Type1 was allotted to the head when AOB of Type2 is arranged just before that Type2. Drawing 32 (e) is drawing showing the arrangement pattern with which AOB of Type2 and Type2 is allotted to the termination of the truck to precede, and AOB of Type1 is allotted to the head of the truck which follows. Since this also fulfills conditions 2, it can be unified on one truck. As mentioned above, only when it judges beforehand whether two conditions which two trucks which should be integrated mentioned above are fulfilled in integration of a truck and judged with fulfilling these two conditions, two trucks are unified to one.

[0128] Then, the renewal of TKI at the time of dividing a truck (case4) is explained.

{17-5_22-19_33A, B} TKI setting-out drawing 33 in the case of dividing a truck (a) and (b) are drawings supposing the case where one truck is divided into two trucks. TrackManager in this Fig. shall be the same as that of TrackManager shown in drawing 27 , and the operator shall wish edit of dividing TrackC into two trucks called TrackC-TrackF, in this Fig. If it is going to divide TrackC into TrackC-TrackF, AOB002.SA1 corresponding to TrackF will be generated. In drawing 33 (a), TKI#2 which TKI#2 are set as "Unused", and are set as "Unused" as a result of division as shown in drawing 33 (b) are assigned to newly generated AOB002.SA1.

[0129] {-- the renewal of 17-5_22-19_33A, B-1_34A, B} directory entry, and a FAT value -- in case AOB003.SA1 is divided here and AOB002.SA1 is generated, a directory entry and a FAT value must be updated. It explains below how these directory entries and a FAT value are updated. Drawing 34 (a) is drawing showing how the SD_Audio directory entry about the SD_Audio

directory where AOB003.SA1 belongs before division is described. It is divided into plurality and AOB003.SAs1 are Clusters 007, 008, 009, and 00A.... It shall be stored in 00D and 00E. In this case, "the cluster number of the file beginning" is described to be "007" about AOB003.SA1 in a directory entry, and they are Clusters 007, 008, 009, and 00A.... They are the FAT values 007, 008, 009, and 00A corresponding to 00D.... 00D is described to be (009), ... (00A) (00D), and (00E), respectively (008).

[0130] When dividing the second half section of AOB003.SA1 in this condition and obtaining AOB002.SA1, the "file name" about AOB002.SA1, a "file extension child", and "the cluster number of the file beginning" are added to an SD_Audio directory entry. Drawing 34 (b) is drawing showing how the SD_Audio directory entry about the SD_Audio directory where AOB003.SA1 belongs after division is described.

[0131] Cluster 00F in this Fig. store the copy of the content of cluster 00B including the division boundary specified by the operator. The division part which follows the division part of AOB002.SA1 stored in cluster 00B is stored after Clusters 00C and 00D and 00E. The head part of AOB002.SA1 is stored in cluster 00F. The remaining part Since it is stored after Clusters 00C and 00D and 00E, to "the cluster number of the file beginning" about AOB002.SA1 Cluster number 00F which show cluster 00F are described, and (00C), (00D), and (00E) are described by the FAT values 00F, 00C, 00D, and 00E matched with Clusters 00F, 00C, 00D, and 00E.

[0132] {17-5_22-19_33A, B-2_35A, B} After obtaining AOB002.SA1 by renewal of the directory entry beyond setting out of the information element in TKI, and a FAT value, it explains how the information element in TKI about AOB002.SA1 is set up. When generating TKI about the divided truck, two kinds, what to copy what is indicated by the original TKI and inherit (1), and the thing (2) which must be updated based on the original TKI, exist in the information element of TKI. TKTXTI_DA and ISRC correspond to the former and the remaining components which make BIT and TKTMSRT the start correspond to the latter. With this operation gestalt, since these both exist, in case TKI about the divided truck is generated, while copying TKI of a dividing agency and creating the new chicken type of TKI, division and updating are

performed for TKTMSRT and BIT which are contained in it, and the procedure of updating the remaining information element is made.

[0133] Drawing 35 (a) is drawing supposing the case where AOB is divided by AOB_FRAME of arbitration. In this Fig., the 1st step shows AOB_ELEMENT#1 which is four AOB_ELEMENT, AOB_ELEMENT#2, AOB_ELEMENT#3, and AOB_ELEMENT#4. Each data length of these four AOB_ELEMENT is set as TKTMSRT as four TMSRT_entry#k -1, #k, #k+1, and #k+2 (referred to as k= 2 here). In this Fig., supposing the division boundary bd1 is set up in AOB_ELEMENT#2, AOB_ELEMENT#2 will be divided into field ** which consists of a front frame from the division boundary bd1, and field ** which consists of a back frame from the division boundary bd1. Drawing 35 (b) is drawing showing the condition that AOB was divided in the part in the middle of AOB_ELEMENT#2, and two AOB(s) called AOB#1 and AOB#2 were obtained.

[0134] {17-5_22-19_33A, B-3_36} Setting-out drawing 36 of BIT is drawing showing how BIT is set up, when AOB is divided, as shown in drawing 35 . AOB shown in drawing 35 is divided on the division boundary bd1, and it turns out that AOB#1 obtained by the division contains three AOB_ELEMENT called AOB_ELEMENT#1, AOB_ELEMENT#2, and AOB_ELEMENT#3 in AOB#2 including two AOB_ELEMENT called AOB_ELEMENT#1 and AOB_ELEMENT#2.

[0135] Although the triangular pennant-like notation is also given to each of these AOB_ELEMENT, these show that TMSRT_entry contained in TKI corresponding to AOB, respectively is set up. AOB#1 first obtained by division is explained. Since AOB_ELEMENT#1 contained in AOB#1 and AOB_ELEMENT#2 occupy a cluster 007 - cluster 00A, AOB#1 is treated considering a cluster 007 - cluster 00A as one unit. Since AOB_ELEMENT#2 in AOB#1 do not occupy even the termination of cluster 00A here and even the division boundary bd1 where cluster 00A exists is occupied, SZ_DATA about AOB#1 will direct the data length from the field md0 to the division boundary bd1 in cluster 00A. Although "FNs_1 st_TMSRTE" of AOB#1 is not different from before division, "FNs_Last_TMSRTE" of AOB#1 differs point's of directing frame number's from head before division of AOB_ELEMENT#2 to

division boundary bd's1 division before.

[0136] Then, AOB#2 obtained by division are explained. AOB_ELEMENT#1 contained in AOB#2, AOB_ELEMENT#2, and AOB_ELEMENT#3 occupy cluster 00B- cluster 00F. In cluster 00F, it is the cluster which stores the copy of the content of cluster 00A (the reason for storing the copy of cluster 00A in cluster 00F is that it is necessary to assign this cluster and a different cluster to AOB_ELEMENT#1 contained in AOB#2 since cluster 00A is occupied by AOB_ELEMENT#2 of AOB#1.).

[0137] Middle [in cluster 00E], since AOB_ELEMENT#1 in AOB#2 is not occupied from the head of cluster 00F and it occupies the cluster 00 division boundary bd1 or subsequent ones where F exists, SZ_DATA about AOB#2 will direct the sum of the data length to a part, and the data length which AOB_ELEMENT#1 occupies in cluster 00F from the head of cluster 00B.

[0138] AOB_ELEMENT#2 of AOB#1 are recorded, the copy of cluster 00A stored in cluster 00F must except the part occupied by AOB_ELEMENT#2 of AOB#1 from AOB#2, and the size in which DATA_Offset about BIT of AOB#2 is occupied by AOB_ELEMENT#2 of AOB#1 in cluster 00F is set to it.

[0139] As this drawing also shows, in division of AOB, only AOB_ELEMENT including a division boundary is divided into two, and it turns out that AOB_ELEMENT before and behind that division boundary is not changing from the thing before division. Therefore, "FNs_Last_TMSRTE" of AOB#2 is set as the same value as "FNs_Last_TMSRTE" of AOB_ELEMENT#4 before division, and the frame number by which "FNs_1 st_TMSRTE" of AOB#2 is contained in a part for the trailer after the division boundary in AOB_ELEMENT#2 before AOB_ELEMENT#1, i.e., division, of AOB#2 is set up.

[0140] {17-5_22-19_33A, B-4_37} Setting-out drawing 37 of BIT is drawing showing still more concretely how BIT changes before and after division. BIT on the left-hand side of drawing 37 shows the example of setting out of BIT before division. Data_Offset is set as X, SZ_DATA is set up with "52428" and, as for BIT before dividing a truck, TMSRTE_Ns is set up with the "n" individual. FNs_1 st_TMSRTE is set as "94 frames" about "80 frames" and FNs_Middle_TMSRTE, and it turns out that FNs_Last_TMSRTE is set as "50

frames."

[0141] Setting out of BIT about two trucks after division is shown in the right-hand side of drawing 37 . As AOB corresponding to Book BIT shows drawing 35 (a), when it is divided, although Data_Offset is set as the same value "x" as division before in BIT of 1 truck eye, it is updated by the data length "Q" to the dividing point bd1 at SZ_DATA, and is updated by "k pieces" which are the numbers of TMSRT_entry from 1st TMSRT_entry to k-th TMSRT_entry at TMSRTE_Ns. Like [FNs_Middle_TMSRTE / FNs_1 st_TMSRTE and] division before, although set as 80 or 94 frames, since p AOB_FRAME is contained in AOB_ELEMENT of the last of AOB of 1 truck eye after division in drawing 35 (a), FNs_Last_TMSRTE is set as "p frames."

[0142] Data_Offset is set as R, SZ_DATA is set up with the data length "Q" to the point bd1 of an original copy dividing [SZ#DATA52428-], and, as for BIT of 2 truck eye, TMSRTE_Ns is set up with +one n-k (it is a number adding the n-k individual which is the TMSRT_entry number from k-th TMSRT_entry to n-th TMSRT_entry, and one piece which is the number of k-th newly because of division added TMSRT_entry.). Like [FNs_Last_TMSRTE / FNs_Middle_TMSRTE and] division before, although set as 94 or 50 frames, since 94-p AOB_FRAME is contained in AOB_ELEMENT of the beginning of AOB of 2 truck eye after division, FNs_1 st_TMSRTE is set as "94-p frames."

[0143] {17-5_22-19_33A, B-5_38} Setting-out drawing 38 of BIT is drawing showing TKTMSRT after division. About TMSRT, it is as follows first. TMSRT of 1 truck eye contains (TMSRT_entry#1 - TMSRT_entry#k) from the start of TMSRT of AOB before division to the k-th entry. here -- being careful -- since it is only that AOB_ELEMENT#k including a division boundary contains field **, only the data size of the part in which this k-th entry is equivalent to this field ** is contained. TMSRT of 2 truck eye contains (TMSRT_entry#k-TMSRT_entry#n) from the k-th entry before division to the n-th entry. here -- being careful -- since AOB_ELEMENT#k including a division boundary is only that field ** is included in 2 truck eye, only the data size of the part in which the k-th entry before division is equivalent to this field ** is contained.

[0144] If division and updating are performed for TKTMSRT and BIT and the remaining information element is updated while copying TKI, TKI about the

new truck obtained by division will be obtained. The truck corresponding to an AOB file can be divided into two with the condition of having been enciphered, without decrypting the enciphered AOB file like the case of integration. Since decode and re-encryption do not follow in the case of AOB file division, it turns out that the processing load at the time of dividing a truck is mitigated. Thereby, a truck can be edited even when the processing engine performance of a regenerative apparatus is low.

[0145] Although it became a long sentence above, the explanation about TKI is ended. Then, a play list is explained.

{17-6} Playlistmanager shown in Playlistmanager drawing 17 PlaylistManager_Information which manages the play list stored in flash memory card 31 as shown in the outgoing line h5 of a broken line (PLMGI), Default_Playlist_Information (DPLI) which manages all the trucks stored in flash memory card 31, PlaylistInformation(PLI) #1, #2, #3, #4, #5 ..... It consists of n. # Default_Playlist information As shown in the outgoing line h6 of a broken line, Default_Playlist_General_Information (DPLGI), Default_Playlist_Track_Serch_Pointer(DPL_TK_SRP) #1, #2, #3, #4 .... #m shows becoming. Moreover, each PLI is Playlist_General_Information (PLGI), Playlist_Track_Serch_Pointer(PL_TK_SRP) #1, #2, #3, and #4, as shown in the outgoing line h7 of a broken line.... #m shows becoming.

[0146] The difference between Default_Playlist information and PlayList information is explained here. To Default_Playlist information being obliged to specify all trucks, such a duty does not exist but PlayList information should just specify the truck of arbitration. Therefore, it is suitable for the application of making flash memory card 31 memorizing, and a regenerative apparatus generating automatically PlayList information which specifies only the truck of a predetermined genre among two or more trucks which generate PlayList information as which the user specifies only favorite his own truck, and are memorized by flash memory card 31, and storing it in flash memory card 31.

[0147] {17-7_18} When the number of a play list and data size drawing 18 are referred to, the maximum number of a play list is 99 pieces. Moreover, Playlist Manager Information (PLMGI) and Default Playlist Information (DPLI) are 2560 bytes of fixed lengths in total. Playlist Information (PLI) is also 512 bytes

of fixed length. DPL_TK_SRP contained in Default_Playlist information contains DPL_TK_ATR and DPL_TKIN. On the other hand, PL_TK_SRP contained in PlayList information contains only PL_TKIN. These DPL_TK_ATR, DPL_TKIN, and PL_TKIN have the format shown in drawing 39 .

[0148] {17-8_39-1} Format drawing 39 (a) of DPL_TK_SRP is drawing showing a format of DPL_TK_SRP. In drawing 39 (a), DPL_TKIN is described by the 9th bit from the 0th bit, DPL_TK_ATR is described by 15 bits from the 13th bit, and DPL_TK_SRP is secured to reservation from the 10th bit to 12 bits (reserved).

[0149] Next, a TKI number is described by DPL_TKIN which occupies the field from the 0th bit to the 9th bit. By describing a TKI number here, it becomes possible to specify TKI.

{17-9_39B} Format drawing 39 (b) of PL_TK_SRP is drawing showing a format of PL_TK_SRP. PL_TK_SRP has the field from the 0th bit to the 9th bit, and PL_TKIN, i.e., a TKI number, is described here.

[0150] {17-8_39A-2} The example of setting out of DPL_TK_ATR is shown within the limit pulled out by the arrow heads h51 and h52 of a broken line from "DPL_TK_ATR" of the block diagram 39 of DPL_TK_ATR (a). Setting out of DPL_TK_ATR about DPL_TK_SRP is the same as setting out of TKI_BLK_ATR about TKI, and it is set up any of "Track", "Head_of_Track", "Midpoint_of_Track", and "End_of_Track" they are so that he can understand also from this publication within the limit.

[0151] When TKI specified in TKIN is specifically using it and the audio object corresponding to one truck is recorded on the AOB file corresponding to the TKI concerned ("Track" in TKI_BLK_ATR of TKI), as for DPL_TK_ATR, the value of "000b" is set up. When TKI specified in TKIN is using it and the audio object only corresponding to the head section of a truck is recorded on the AOB file corresponding to the TKI concerned ("Head_of_Track" in TKI_BLK_ATR of TKI), as for DPL_TK_ATR, the value of "001b" is set up.

[0152] When TKI specified in TKIN is using it and the audio object only corresponding to the pars intermedia of a truck is recorded on the AOB file corresponding to the TKI concerned ("Midpoint_of_Track" in TKI_BLK_ATR of

TKI), the value of "010b" is set to DPL_TK_ATR. When TKI specified in TKIN is using it and the audio object only corresponding to the trailer of a truck is recorded on the AOB file corresponding to the TKI concerned ("End_of_Track" in TKI_BLK_ATR of TKI), the value of "011b" is set to DPL_TK_ATR.

[0153] TKI specified in TKIN is intact, and when it is TKI to which only the field of TKI is secured and which was case [TKI] namely, deleted ("Unused" in TKI_BLK_ATR of TKI), the value of "100b" is set up. TKI specified in TKIN is intact, and when the field of TKI is not secured (i.e., when it is TKI of an initial state), the value of "101b" is set up.

[0154] "DPL_TK_SRP" has response relation with which thing among two or more TKI(s) by describing the number of TKI to DPL_TKIN. Moreover, the ranking of DPL_TK_SRP in Default_Playlist information shows whether DPL_TK_SRP and AOB (AOB file) corresponding to TKI which has response relation are reproduced by what position. by these things, the sequence of DPL_TK_SRP in Default_Playlist information reproduces two or more trucks in what kind of sequence, or defines the playback sequence of a truck -- things -- **

[0155] {17-9_40-1} Correlation drawing 40 of Default_Playlist information, TKI, and an AOB file is drawing showing the correlation of Default_Playlist information, TKI, and an AOB file. The 2nd in this Fig., the 3rd, and the 4th step are the same as the 1st step of drawing 19 , the 2nd step, and the 3rd step, and TrackManager and eight AOB files containing eight TKI(s) are shown. Differing from drawing 19 is the point that the rectangular-head frame which shows Default_Playlist information to the 1st step is described. Eight reels contained in the frame of the 1st step show eight DPL_TK_SRP contained in Default_Playlist information. The upper case of these reels shows DPL_TK_ATR and the lower berth shows DPL_TKIN.

[0156] the arrow heads DT1, DT2, DT3, and DT4 in this Fig. -- if ..... is referred to -- between DPL_TK_SRP#1 and TKI#1 -- response relation -- being materialized -- **** -- DPL_TK_SRP#2 and TKI# -- it turns out that response relation is materialized also among 2, between DPL_TK_SRP#3 and TKI#3, and between DPL_TK_SRP#4 and TKI#4. Furthermore, if DPL_TK_ATR in each DPL_TK_SRP is referred to, each of DPL_TK_SRP#1,

DPL_TK_SRP#2, DPL_TK_SRP#3, and DPL_TK_SRP#8 is set up with Track. four [namely,] called DPL_TK_SRP#1 ->TKI#1 (AOB001.SA1), DPL_TK_SRP#2 ->TKI#2 (AOB002.SA1), DPL_TK_SRP#3 ->TKI#3 (AOB003.SA1), and DPL_TK_SRP#8 ->TKI#8 (AOB008.SA1) -- constructing -- the truck with which each became independent is supported.

[0157] As for DPL_TK_ATR in DPL_TK_SRP#7, it turns out that no DPL_TK_ATR of DPL_TK_SRP#4, DPL_TK_SRP#5, DPL_TK_SRP#6, and DPL_TK_SRP#7 is set up with Track, but DPL_TK_ATR in DPL_TK_SRP#4 is set up with "Head_of_Track", and "End_of_Track", DPL_TK_SRP#5, and DPL_TK_SRP#6 are set up with "Midpoint_of_Track." As for this, DPL_TK_SRP#4 and TKI#4 (AOB004.SA1) which have response relation mean that TKI#7 (AOB007.SA1) in which TKI#5 (AOB005.SA1) which is the head section of a truck and has DPL_TK_SRP#5, #6, and response relation, and TKI#6 (AOB006.SA1) have the pars intermedia of a truck, DPL_TK_SRP#7, and response relation is the trailer of a truck.

[0158] The sequence of DPL_TK_SRP in DefaultPlaylist shows [each] in what kind of sequence AOB matched with TKI is reproduced. DPL_TK_SRP#1 in DefaultPlaylist of this Fig., #2, #3, #4 ..... DPL_TKIN of 8 # TKI#1, #2, #3, #4 ..... Since 8 is shown, # An arrow head (1), (2), (3), (4) ..... As shown in (8), AOB001.SA1 corresponding to TKI#1 is reproduced by the 1st. AOB004.SA1 corresponding to the 3rd and TKI#4 in AOB003.SA1 corresponding to the 2nd and TKI#3 in AOB002.SA1 corresponding to TKI#2 will be reproduced by the 4th.

[0159] {17-10_41} Example drawing 41 of setting out of DefaultPlaylist and PlayList information is drawing having shown the example of setting out of DefaultPlaylist and PlayList information with the same notation as drawing 40 . The rectangular-head frame in the 1st step in this Fig. shows Default_Playlist information, and three rectangular-head frames in the 2nd step show PlayList information. The reel which the reel contained in DefaultPlaylist shows eight DPL_TK_SRP contained in DefaultPlaylist, and is contained in PlayList information shows three or four PL_TK_SRP. Setting out of TKIN of each DPL_TK_SRP contained in the Default_Playlist information on this Fig. is the same as that of drawing 40 . However, it turns out that setting out of TKIN of

PL_TK_SRP contained in PlayList information completely differs from it of DPL_TK_SRP.

[0160] {17-10_42} Response drawing 42 of DPL_TK_SRP and TKI is drawing showing a response with DPL_TK_SRP and TKI using the same notation as drawing 40 . Playlist#1 consists of PL_TK_SRP#1, #2, and #3 in drawing 42 . Among these, PL_TKIN of PL_TK_SRP#1 is indicated to be #3, and since #1 and PL_TKIN of PL_TK_SRP#3 are indicated to be #2, when PL_TKIN of PL_TK_SRP#2 reproduces a truck using PlayList information #1, as shown in an arrow head (11), (12), and (13), two or more AOB(s) are reproduced in order of AOB#3, #1, and #2.

[0161] Playlist#2 consist of PL_TK_SRP#1, #2, and #3. Among these, since PL_TKIN of PL_TK_SRP#1 is indicated to be #8 and PL_TKIN of PL_TK_SRP#2 and #3 is indicated to be #3 and #1, when reproducing a truck using PlayList information #2, as shown in an arrow head (21), (22), and (23), two or more AOB(s) are reproduced in the sequence of AOB#8, #3, and #1, i.e., completely different sequence from Playlist#1.

[0162] Playlist#3 consist of PL_TK_SRP#1, #2, #3, and #4. Among these, since PL_TKIN of PL_TK_SRP#1, #2, #3, and #4 is indicated to be #8, #4, #3, and #1, when reproducing a truck using PlayList information #3, AOB is reproduced in order of the playback shown below. As first shown in an arrow head (31), AOB#8 which constitute TrackE are reproduced, and AOB#4 which constitute TrackD as shown in an arrow head (32), AOB#5, AOB#6, and AOB#7 are reproduced following this. Then, it is reproduced in the sequence of AOB#3 which constitute TrackC and TrackA as shown in an arrow head (33) and (34), and AOB#1. here -- being careful -- when a truck consists of two or more TKI(s), it is the point that only two or more top TKI numbers are described by the entry of PL_TK_SRP among TKI(s). Although DPL_TK_SRP in Default_Playlist information specified TKI#4 which are four TKI(s) about TrackD, TKI#5, TKI#6, and TKI#7 when said concretely, PL_TK_SRP in PlayList information does not need to specify these four TKI(s). That PL_TK_SRP#2 of Playlist#3 specify only TKI#4 among TKI#4-TKI#7 means this.

[0163] On the other hand, DPLI containing two or more DPL_TK_SRP has

data size which is settled in 1 sector, and resides permanently on RAM. Therefore, when reproducing each truck based on Playlist, each thing for which TKI is searched at a high speed becomes possible by referring to DPL_TK_SRP which resides permanently on RAM. That is, in order to reproduce TKI (AOB) using PL_TK_SRP only two or more top TKI numbers are described to be among TKI(s), DPL_TK_SRP which resides permanently on RAM based on TKI described by PL_TK_SRP is searched, and it judges whether the truck consists of two or more TKI(s). When it consists of two or more TKI(s), it passes through the procedure of reproducing all corresponding TKI(s) (AOB).

[0164] As mentioned above, DefaultPlaylist and two or more PlayList information are described by PlayListManager, and if the playback sequence which is different from each other in DPL_TKIN of DPL_TK_SRP and PL_TK_SRP which constitute these, and PL_TKIN, respectively is indicated, two or more AOB(s) will be reproduced in order of the playback which is different from each other, respectively. If reproduced in order of completely different playback, an operator can use flash memory card 31 with sensation in which two or more music albums are stored.

[0165] moreover -- being careful -- it is the point that the data size of DPL_TK_SRP is small among DPL_TK_SRP matched with the AOB file by each, and TKI (it is only 2 bytes), and the data size of TKI is large (there are no less than 1024 bytes.). although access to flash memory card 31 occurs mostly, even if replacing the sequence of TKI in TrackManager replaces the sequence of DPL_TK_SRP in Default_Playlist information and PlayList information, access to flash memory card 31 does not have 7 so mostly. While navigation data change the sequence of DPL_TK_SRP in DefaultPlaylist positively in the time of that edit in view of this point according to editing operation, he is trying to maintain the sequence of TKI in TrackManager uniformly irrespective of editing operation.

[0166] {17-9_40-2_43A, B} By replacing the sequence of DPL_TK_SRP, next replacing the sequence of DPL_TK_SRP in Default_Playlist information explains how editing operation of changing the playback sequence of a truck is performed. Drawing 43 (a) and (b) are drawings supposing the case where

the sequence of a truck is replaced. Setting out of DPL_TK_SRP in drawing 43 (a) and TKI is the same as drawing 40 . DPL_TKIN [in / on drawing 40 (a) and / DPL_TK_SRP#3] replaces the sequence of DPL_TK_SRP#8 with DPL_TK_SRP#3 enclosed with a thick frame in this condition, although DPL_TKIN in TKI#3 and DPL_TK_SRP#8 was set up with TKI#8. (1) in drawing 43 (b), (2), (3), (4), (5), (6), (7), and (8) show the playback sequence of the truck after sequence exchange. Although the playback sequence in drawing 43 (a) is TrackA, TrackB, TrackC, TrackD, and TrackE when this is cared about, since the sequence of DPL_TKIN about DPL_TK_SRP#3 and DPL_TK_SRP#8 was replaced, for the Default_Playlist information in drawing 43 (b), it will be reproduced in order of TrackA, TrackB, TrackE, TrackD, and TrackC. Thus, the playback sequence of a truck can be simply changed by replacing the sequence of DPL_TK_SRP in Default_Playlist information.

[0167] In the place explaining editing operation called modification actuation of a truck, like the case of TKI When some trucks were deleted (case1), after some trucks were deleted, [when dividing one truck when recording a new truck (case2) and unifying two of two or more trucks of arbitration on one truck (case3), and obtaining two trucks (case4)] It explains how DPL_TK_SRP and TKI are updated.

[0168] {17-9_40-3_44A, B} When deleting a truck, the case (case1) where some trucks are deleted is explained first. Drawing 44 (a) and (b) are drawings showing how DefaultPlaylist, TrackManager, and an AOB file are updated, when deleting DPL_TK_SRP#2 and TKI#2 among DefaultPlaylist(s) shown in drawing 40 . Drawing 44 has the same part as drawing 27 quoted by explanation of deletion of TKI. That is, the 2nd in drawing 44 , the 3rd, and the 4th step are the same as that of drawing 27 . Differing is the point that the Default_Playlist information which contains two or more DPL_TK_SRP in the 1st step is indicated like drawing 40 . The user should delete TrackB which consists of DPL_TK_SRP#2 ->TKI#2 (AOB002.SA1) enclosed with the thick frame in drawing 44 (a). In this case, DPL_TK_SRP#2 are deleted in Default_Playlist information, and as the field which DPL_TK_SRP#2 occupied packed in DPL_TK_SRP#3-DPL_TK_SRP#8, sequence advances one [at a time]. Thus, the sequence of each DPL_TK_SRP is advanced and

DPL_TK_SRP#8 of the very end are set as "Unused." On the other hand, migration which TKI is [only being set as "Unused" and] as explained using drawing 27 (a) and (b), and packs TKI#2 is not performed. Moreover, it turns out that AOB002.SA1 is deleted. Although advance of sequence was performed about DPL_TK_SRP, since advance of sequence is not performed about TKI, DPL_TKIN in DPL_TK_SRP is updated by drawing 44 (b). Namely, new DPL_TKIN of DPL_TK_SRP#2 is directing TKI#3, as shown in an arrow head DT 11, DPL_TKIN of DPL_TK_SRP#4 directs TKI#4 and, as for DPL_TKIN of TKI#5 and DPL_TK_SRP#5, DPL_TKIN of DPL_TK_SRP#3 is directing TKI#6, respectively, as shown in an arrow head DT 12. Furthermore, as shown in an arrow head DT 13, as for DPL_TKIN of DPL_TK_SRP#8 set as "Unused", it turns out that TKI#2 set as "Unused" are set up.

[0169] Although DPL_TK_SRP which is under activity is advanced to a head when deletion of a truck is performed, while TKI corresponding to it had maintained arrangement of a basis, it turns out that it is set up intact. Thus, since arrangement of TKI is made into immobilization before and after edit, the processing load accompanying edit processing is mitigable.

{17-9_40-4_45A, B} After assignment of TKI in the case of recording a truck, then some trucks are deleted, the case (case2) where a new truck is recorded is explained. Drawing 45 (a) and (b) are drawings showing how the writing is performed, when TKI of "Unused" and DPL_TK_SRP exist and it writes in new TKI and DPL_TK_SRP here. In drawing 45 (a) and (b), when the case which assigns new TKI to TKI of "Unused" is explained, it has the same part as quoted drawing 28 (a) - (b). That is, the 2nd in drawing 45 (a) and (b), the 3rd, and the 4th step are the same as the 1st of drawing 28 (a) and (b), the 2nd, and the 3rd step. Differing is the point that the Default_Playlist information which consists of DPL_TK_SRP of the plurality [step / 1st] of drawing 45 is described. In drawing 45 (a), DPL_TK_SRP#4-DPL_TK_SRP#8 are "Unused" and, on the other hand, it turns out that TKI#2, TKI#4, TKI#5, TKI#7, and TKI#8 are "Unused(s)" as shown in drawing 28 (a). As it mentioned above that DPL_TK_SRP of "Unused" was summarized in Default_Playlist information to TKI of "Unused" existing in the shape of vermin in TrackManager, it is because, as for TKI, such advance is not performed to advance of

DPL_TK_SRP other than "Unused" being performed, as for DPL_TK_SRP.
[0170] The case where it is going to write in TrackD which consists of four AOB(s) here is assumed. TKI about each of the four AOB(s) is written in each of TKI#2, TKI#4, TKI#7, and TKI#8 which is set as "Unused" in TrackManager. On the other hand, DPL_TK_SRP about these four AOB(s) is written in DPL_TK_SRP#4-DPL_TK_SRP#7 in Default_Playlist information. Since these four AOB(s) constitute one truck, "Midpoint_of_Track" and DPL_TK_ATR about DPL_TK_SRP#7 are set [DPL_TK_ATR about DPL_TK_SRP#4] up for "Head_of_Track" and DPL_TK_ATR about DPL_TK_SRP#5 and DPL_TK_SRP#6 with "End_of_Track."
[0171] Moreover, DPL_TKIN about DPL_TK_SRP#4 is set up with TKI#2 and DPL_TKIN about TKI#7 and DPL_TK_SRP#7 is set [DPL_TKIN about DPL_TK_SRP#5] up for DPL_TKIN about TKI#4 and DPL_TK_SRP#6 with TKI#8. TKI#2, TKI#4, TKI#7, and TKI#8 are managed as the 4th truck TrackD by setting out of above DPL_TKIN and DPL_TK_ATR -- things -- **
[0172] In the above processing, although the writing to TKI of "Unused" was performed, the point that no fluctuation is made about TKI#1, TKI#2, TKI#3, and TKI#4 is the same as that of the case of drawing 28 .
{17-9_40-5_46A, B} The renewal of Default_Playlist information at the time of continuing about the case where a truck is unified (case3), and unifying a truck (case3) is explained. Drawing 46 (a) and (b) are drawings supposing the case where a truck is unified. This Fig. has the same part as drawing 29 (a) quoted when integrated processing of TKI was explained, and (b). That is, the 2nd in drawing 46 (a) and (b), the 3rd, and the 4th step are the same as the 1st step in drawing 29 (a) and (b), and the 2nd step. Difference points are TKI#2 which Default_Playlist information is indicated, and DPL_TK_SRP#8 contained in it are set as "Unused", and are similarly set as "Unused", and a point of having response relation, in drawing 46 (a) and (b). In this Fig., if integrated processing of a truck as shown in drawing 29 is made to an AOB file and TKI, it will shift every one content of DPL_TK_SRP#3-DPL_TK_SRP#6, and will copy the content of description of DPL_TK_SRP#7 enclosed with a thick frame to DPL_TK_SRP#3. About TKI, the same update process as the case where it is shown in drawing 29 is made.

[0173] {17-9_40-6_47A, B} The renewal of Default_Playlist information at the time of continuing about the case where a truck is divided (case4), and dividing a truck (case4) is explained. Drawing 47 (a) and (b) are drawings supposing the case where a truck is divided. This Fig. has the same part as drawing 33 (a) quoted when the division processing about TKI was explained, and (b). That is, the 2nd step in this Fig. and the 3rd step are the same as the 1st step in drawing 33 (a) and (b), and the 2nd step. Difference points are TKI#2 which Default_Playlist information is indicated, and DPL_TK_SRP#8 contained in it are set as "Unused", and are similarly set as "Unused", and a point of having response relation, in drawing 47 (a) and (b). In this condition, if it is going to divide TKI#3 and AOB003.SA1 enclosed with a thick frame into two like the case of drawing 33 , every one sequence of DPL_TK_SRP#3-DPL_TK_SRP#7 will be carried down and DPL_TK_SRP of "Unused" in Default_Playlist information will be moved to DPL_TK_SRP#3. TKI#2 obtained by division are matched with DPL_TK_SRP#3 after migration. Although AOB002.SA1 matched with TKI#2 stores the second half section of AOB003.SA1 from the first, DPL_TK_SRP#2 exist before DPL_TK_SRP#3 matched with TKI#2, and, as for these DPL_TK_SRP#2, TKI#2-AOB002.SA1 is matched. Namely, although AOB002.SA1 and AOB003.SA1 store a part for the second half part of original AOB003.SA1, and the first portion DPL_TK_SRP#2 which specify these, and DPL_TK_SRP#3 Since playback sequence is specified that it reproduces these AOB files in order of AOB003.SA1 and AOB002.SA1 a part for the second half part of original AOB003.SA1 and the first portion is reproduced by playback sequence assignment of DPL_TK_SRP in order of a part a part for the first portion, and the second half -- things -- **

[0174] {17-9_40-8} By combining four editing operation beyond application of edit processing, an operator can perform various editing operation. That is, if the announcement part is divided as a truck of a piece and the truck is deleted after that by division processing of the above-mentioned truck for a D.J.'s announcement to get down from close to the head part of a certain truck, and delete this, only a D.J.'s announcement can be deleted selectively.

[0175] The regenerative apparatus constituted in order to finish the

explanation about navigation data above, then to reproduce such navigation data and presentation data is explained.

{48-1} The external view 48 of a regenerative apparatus is drawing showing the regenerative apparatus of the pocket mold about the flash memory card 31 concerning this operation gestalt. In this Fig., the regenerative apparatus has the key panel for receiving from an operator key strokes, such as insertion opening with which flash memory card 31 is inserted, playback, forward-search playback, hard flow search playback, a rapid traverse, rewinding, and a halt, and the liquid crystal display, and has the same appearance as the usual pocket mold audio equipment. The Playlist key which receives selection of a play list / truck on a key panel, The "|<<key", "> which receives skip at head of degree truck> | key" which receives a skip at the head of a truck, ">> key", "<< key" which receive a rapid traverse, rewinding, forward-search playback, and hard flow search playback, The Display key which receives the actuation on which a still picture is displayed when the still picture is stored in flash memory card 31, The Rec key, Stereo/Monoral selection which receive sound recording actuation from an operator, It has the Edit key which receives edit of the Audio key which receives sampling frequency selection from an operator, the Mark key which receives assignment of a bookmark, and a truck, and a title input.

[0176] {48-2} It is following improving point (1) - (4) that the pocket mold regenerative apparatus of the flash memory card 31 of amelioration **** in the pocket mold regenerative apparatus of flash memory card 31 differs from the usual pocket mold audio equipment. In order to receive assignment of Default_Playlist information, PlayList information, and a truck from an operator, namely, to a liquid crystal display The inside of thing (1) by which a play list and the list display of a truck are made and the play list by which it was indicated by the list such, or a truck, With playback progress of the thing (2) and the truck with which the key assignment for making the thing of arbitration specify as an object for playback for edit is made, to a liquid crystal display In case thing (3) and the time search function as which the playback progress time of day which is a truck is displayed, and division edit are performed, it is thing (4) in which the jog dial used in order to set up playback start time is

prepared.

[0177] {48-2_49_50} The detail of the point (2) of an improving point (2) improving [detail] is as follows. Drawing 49 is drawing showing an example of the content of a display of the liquid crystal display at the time of selection of a play list being performed, and drawing 50 is drawing showing an example of the content of a display of the liquid crystal display at the time of selection of a truck being performed. "DEFAULTPLAYLIST" in drawing 49 , "PLAYLIST#1", "PLAYLIST#2", "PLAYLIST#3", and "PLAYLIST#4" are ASCII-character trains which show the default play list stored in flash memory card 31, and four play lists. Moreover, "TRACK#1" in drawing 50 (a), "TRACK#2", "TRACK#3", "TRACK#4", and "TRACK#5" are ASCII-character trains which show five trucks where playback sequence is specified by the default play list stored in flash memory card 31. It is shown that these play lists and trucks that attached hatching by drawing 49 and drawing 50 (a) are specified as an object for playback for edit. Thus, a list indication of the truck where playback sequence is specified to a liquid crystal display by the play list is given, and if the depression of a >>| key is made where TRACK#1 is specified as the object for playback, TRACK#2 under it will be specified as the object for playback among the multiple tracks by which it was indicated by the list as shown in drawing 50 (b). If the depression of a >>| key is made where TRACK#2 are specified as the object for playback, TRACK#3 of the lower berth will be further specified as the object for playback among the multiple tracks by which it was indicated by the list as shown in drawing 50 (c). If the depression of a |<< key is made where TRACK#3 are specified as the object for playback, as shown in drawing 50 (d) among the multiple tracks by which it was indicated by the list, TRACK#2 on one step will be specified as the object for playback. Thus, since which truck is chosen as an object for playback according to the depression of a >>| key and a |<< key, if a playback key is pressed as shown in drawing 50 (e) when which truck is chosen as an object for playback, playback of the truck will be started, and if the Edit key is pressed, the truck will be specified as an object for edit.

[0178] {48-3_51} The detail of an improving point (4), then the detail of an improving point (4) are explained. Drawing 51 is drawing showing the

example of actuation of a jog dial. The revolution actuation by the operator is received by the jog dial, and the playback progress time of day currently displayed on the liquid crystal display is made to fluctuate according to the rotation. For example, as shown in drawing 51 (a), playback start time shall be displayed on the liquid crystal display as "00:00:20." In this case, as shown in drawing 51 (b), supposing a jog dial rotates counter clockwise, playback start time will decrease according to that rotation, and will be set to "00:00:10." Moreover, as shown in drawing 51 (c), supposing a jog dial rotates clockwise, playback start time will increase according to the rotation, and will be set to "00:00:30."

[0179] Thus, it is for specifying the playback time of day of the arbitration in a truck to make playback time amount time amount fluctuate, and if the playback time of day of arbitration is specified and a playback key is pressed by the revolution of a jog dial, it will reproduce AOB from the location specified according to the above {a formula 2} and {a formula 3}. Moreover, in case a jog dial specifies the playback start time of arbitration as a division boundary in division edit, it is used in order to tune a division boundary finely.

[0180] {52-1} It continues about the internal configuration of a regenerative apparatus, and the internal configuration of a regenerative apparatus is explained. Drawing 52 is drawing showing the internal configuration of a regenerative apparatus. The card connector 1 for a regenerative apparatus to connect flash memory card 31 in this Fig., A key panel and the user interface section 2 connected with a jog dial, The liquid crystal display 5 which has RAM3, ROM4, a play list and the list display frame which indicates the truck by list, and the playback progress time-of-day frame with which playback progress time of day is displayed, The LCD driver 6 for driving a liquid crystal display, and the Di scrambler 7 which cancels encryption of AOB_FRAME using different FileKey for every AOB file, If descrambling of AOB_FRAME is performed by the Di scrambler 7 With reference to the ADTS header of the AOB_FRAME concerned, by decoding the AOB_FRAME concerned D/A conversion of the PCM data obtained by decode of the AAC decoder 8 which obtains PCM data, and the AAC decoder 8 is carried out, and it has D/A converter 9 outputted to a loudspeaker through a phones jack, and CPU10

which performs integrated processing in a regenerative apparatus. The new configuration for processing TrackManager and Default_Playlist information is not looked at by this regenerative apparatus so that this hardware configuration may also show. The DPLI resident area 11 secured in RAM3, the PLI storing field 12, the TKI storing field 13, the FileKey storing field 14, the double buffer 15, and the playback control program and edit control program stored in ROM4 are prepared for processing of TrackManager and Default_Playlist information.

[0181] {52-2} The DPLI resident area 11DPLI resident area 11 is a field secured in order to station permanently the Default_Playlist information by which was connected to the card connector 1 and reading appearance was carried out from flash memory card 31.

{52_12} The PLI storing field 12PLI storing field 12 is a field secured since the PlayList information which is chosen by the operator and has become an object for playback is stored.

[0182] {52-3} The TKI storing field 13TKI storing field 13 is a field secured since only TKI corresponding to the AOB file which is an object for playback among two or more TKI(s) contained in TrackManager is stored, and has the data size for one TKI.

{52-4} The FileKey storing field 14FileKey storing field 14 is a field secured since only FileKey corresponding to the AOB file which is an object for playback among two or more FileKey(s) contained in AOBSA1.KEY in a protection field is stored.

[0183] {52-5} Double buffer 15 double buffer 15 is an input output buffer used when performing input process of carrying out a sequential input and storing the cluster data (data stored in per cluster piece) by which reading appearance was carried out from flash memory card 31, and output processing of reading encryption AOB_FRAME from the stored cluster data, and outputting to the Di scrambler 7 one by one to juxtaposition. A double buffer 15 releases the field which the cluster with which the output as AOB_FRAME was able to be managed occupied to a free area one by one, and reserves a partition in partitioning, i.e., the patrol type using a ring pointer, of using this free area for storing of a cluster by which reading appearance

was newly carried out.

[0184] {52-5_53_54A, B} I/O drawing 53 in a double buffer 15 is drawing showing how the data I/O in a double buffer 15 is performed. Drawing 54 (a) and (b) are drawings showing how partitioning of the patrol type which used the ring pointer is performed. In these drawings, the arrow head of the lower left sense shows a pointer, i.e., a store place pointer, about the store place address of cluster data. The arrow head of the upper left sense shows the pointer about the read-out place address of cluster data, i.e., a read-out place pointer. These pointers are used as a ring pointer.

[0185] {54-6_53} If flash memory card 31 is connected to the card connector 1, as shown in arrow heads w1 and w2, reading appearance of the cluster data in the user data area of this flash memory card 31 will be carried out from flash memory card 31. In a double buffer 15, sequential storing of the cluster data by which reading appearance was carried out is carried out in the location shown in the store place pointers WP1 and WP2.

[0186] {52-7_54A} AOB_FRAME which exists in the location directed to read-out place pointer ********** among AOB_FRAME contained in the cluster data stored in this way -- arrow heads r1, r2, r3, r4, and r5 -- as shown in ....., it is outputted to the Di scrambler 7 one by one. the cluster data 002 and 003 store in a double buffer 15 here -- having -- **** -- a read-out place pointer -- ** -- ** -- since it means that reading appearance of all AOB_FRAME contained in a cluster 002 was carried out when it moves as last read-out place shows ******** of drawing 53 , and ** is reached, reading appearance of the cluster 004 is newly carried out, and as shown in the arrow head w6 of drawing 54 (a), the field which the cluster 002 occupied is overwritten.

[0187] {52-8_54B} again -- a read-out place pointer -- ** -- ** -- if it moves as last read-out place shows ****, and ** is reached, since it will mean that reading appearance of all AOB_FRAME contained in a cluster 003 was carried out, reading appearance of the cluster 005 is newly carried out, and as shown in the arrow head w7 of drawing 54 (b), the field which the cluster 003 occupied is overwritten. The above outputs of AOB_FRAME and overwrite of cluster data are repeated repeatedly, and AOB_FRAME contained in an AOB file is outputted to the Di scrambler 7 and the AAC decoder 8 one by one.

[0188] {52-9_55-58} The playback control program stored in the playback control program, then ROM4 which are stored in ROM4 is explained. Drawing 55 is a flow chart which shows the procedure of AOB file read-out processing, and drawing 56 , drawing 57 , and drawing 58 are flow charts which show the procedure of AOB_FRAME output processing.

[0189] {52-9_55-1} In these flow charts, Variable w is a variable which directs two or more each of DPL_TK_SRP, and Variable z is a variable for directing uniquely each AOB file, TKI corresponding to it, and AOB contained in it. Variable y is a variable for directing each AOB_ELEMENT contained in AOB#z directed with Variable z, and is a variable which directs each AOB_FRAME contained in AOB_ELEMENT#y directed with Variable y in Variable x. The procedure of AOB file read-out processing is explained referring to drawing 55 first.

[0190] {52-9_55-2} In step S1, CPU10 reads PlayListManager and indicates Default_Playlist information and the PlayList information by list. CPU10 waits for assignment of according to any of Default_Playlist information and PlayList information to reproduce AOB in step S2. Here, when Default_Playlist information is specified, it shifts to step S3 from step S2, and Variable w is initialized (#w<-1), in step S4, TKI#z specified by DPL_TKIN matched with DPL_TK_SRP#w in Default_Playlist information is specified, and only the TKI#z is read to the TKI storing field 13. And AOB file #z which has the same number as TKI#z in step S5 is specified. It means that the AOB file which should be reproduced was specified at last in the procedure so far. Since it is enciphered, the specified AOB file performs processing of step S6 and step S7 henceforth that encryption of this AOB file should be canceled. That is, at step S6, a protection field is accessed and FileKey#z stored in File Key Entry#z which has the same number as the AOB file #z concerned in a cryptographic key storing file is read. In step S7, CPU10 sets FileKey#z as the Di scrambler 7. Since FileKey was set as the Di scrambler 7, if AOB_FRAME contained in an AOB file is henceforth supplied to the Di scrambler 7 one by one, sequential playback of AOB_FRAME will be carried out by this setting out.

[0191] {52-9_55-3} Each cluster which stores the AOB file is read one by one

henceforth. At step S8, "the cluster number of the file beginning" about the AOB file #z in a directory entry is specified, and CPU10 reads the data stored in the cluster from flash memory card 31 in step S9. At step S10, it judges whether the cluster number is described to be FFF by the FAT value, and if FAT values are values other than FFF, the data stored in the cluster directed with the FAT value in step S11 are read. It shifts to step S10 after this read-out. Here, the data stored in which cluster are read, and when the FAT value matched with the cluster is referred to, as long as which cluster numbers other than FFF are described by the FAT value, processing of the step S10-step S11 is performed repeatedly. Thereby, reading appearance of the cluster directed with the FAT value is carried out one by one. Since it means that reading appearance of all the clusters that constitute AOB file #z was carried out when the cluster number is described to be FFF by the FAT value, it shifts to step S12 from step S10.

[0192] {52-9_55-4} It judges whether variable #w of CPU10 corresponded with the total of DPL_TK_SRP in step S12. If not in agreement, after shifting to step S13 and incrementing variable #w (#w<-#w+1), it shifts to step S4. TKI#z specified by DPL_TKIN#w of DPL_TK_SRP#w in Default_Playlist information in step S4 is specified, and only the TKI#z is read to the TKI storing field 13. Under the present circumstances, although TKI currently used till then is stored in the TKI storing field 13, CPU10 is overwritten using TKI which newly read TKI already stored in the TKI storing field 13. Only the newest TKI will be stored in the TKI storing field 13 by such overwrite. Thus, if TKI is overwritten, processing of step S5 - step S12 will be repeated about AOB file #z. Processing of step S5 - step S12 is repeated, if reading appearance of TKI corresponding to all DPL_TK_SRP contained in Default_Playlist information and the AOB file is carried out, variable #w and the total of DPL_TK_SRP will be in agreement, step S12 will serve as Yes, and this flow chart will be ended.

[0193] {52-9_56_57_58} In parallel to AOB_FRAME output processing or the AOB file read-out processing to cut, CPU10 performs AOB_FRAME output processing according to the flow chart of drawing 56 , drawing 57 , and drawing 58 . In this flow chart, play_time is the time amount in which playback passed until now, i.e., the variable which shows playback progress time of day,

and the content of a display is rewritten at the time of day of a time stamp within the limit of a liquid crystal display 5 according to renewal of this play_time. Moreover, play_data is the data length reproduced until now.

[0194] {52-9_56-1} In step S21, CPU10 is supervising whether the cluster data about AOB file #z were stored in the double buffer 15. Unless cluster data are stored, it carries out by repeating this step S21, but if cluster data are stored, in step S22, initialization of #x and #y will be performed (#x <-1, #y<-1), and AOB_FRAME#x in AOB_ELEMENT#y will be detected from Data_Offset of BIT#z contained in TKI#z or subsequent ones in the cluster about AOB file #z in step S23 after that. As what the ADTS header occupies, with reference to the ADTS header concerned, it analyzes that the data length shown in the ADTS header is audio data of the body section, the ADTS header concerned and the audio data of the body section are read, and SZ_DATA to 7 bytes are outputted to the Di scrambler 7 here. If encryption of AOB_FRAME is canceled by the Di scrambler 7 and decode is performed by the AAC decoder 8, it will be reproduced as voice.

[0195] {52-9_56-2} After detection, in step S24, AOB_FRAME#x is outputted to the Di scrambler 7, only the playback time amount equivalent to AOB_FRAME#x increments playback progress time-of-day play_time in step S25, and only the number of data equivalent to AOB_FRAME#x increments number play_data of reproduced data. Since the playback time amount length of AOB_FRAME is 20msec, 20msec will be added to playback progress time-of-day play_time here.

[0196] If 1st AOB_FRAME is outputted to the Di scrambler 7, it specifies where following AOB_FRAME exists with reference to the ADTS header of AOB_FRAME#x in step S26. At step S27, the increment of variable #x is performed (#x <-#x+1), and following AOB_FRAME is made into AOB_FRAME#x. In step S28, AOB_FRAME#x is supplied to the Di scrambler 7. Then, at step S29, if only the playback time amount equivalent to AOB_FRAME#x increments play_time, only the number of data which corresponds at AOB_FRAME#x will increment play_data to **. After incrementing AOB_FRAME#x, in step S30, CPU10 judges whether #x reached "FNs_1 st_TMSRTE." # Shift to step S26 after checking whether keys

other than the Play key have been pressed in step S31, if x does not reach "FNs_1 st_TMSRTE." Henceforth, processing of step S26 - step S31 is repeatedly performed until #x reach "FNs_1 st_TMSRTE", or until keys other than the Play key are pressed. When keys other than the Play key are pressed here, processing which ends this flow chart and corresponds to the pressed key is performed. It halts, if the key which stopped regeneration and was pressed if the pressed key was a stop key is a halt key.

[0197] {52-9_57-1} On the other hand, if #x reach "FNs_1 st_TMSRTE", step S30 will serve as Yes and will shift to step S32 of drawing 57 . Since all AOB_FRAME contained in AOB_ELEMENT by processing from step S26 to step S30 was supplied to the Di scrambler 7, while incrementing #y in step S32, #x are initialized that a processing object should be shifted to following AOB_ELEMENT (#y<-#y+1, #x <-1).

[0198] Then, in step S33, the start address about AOB_ELEMENT#y is computed with reference to TKTMSRT. Henceforth, processing which consists of step S34 - step S42 is performed. It can be said that processing of step S34 - step S42 is the same as the processing which is the point which is the processing which reads AOB_FRAME contained in AOB_ELEMENT one after another, and consists of step S24 - step S31. The terminating condition of loop-formation processing [in / to #x being reaching "FNs_1 st_TMSRTE" as for the terminating condition of loop-formation processing / in / in differing from processing of step S24 - step S31 / the latter / the former] is that #x reach "FNs_Middle_TMSRTE." # x reaches "FNs_Middle_TMSRTE", and after the loop-formation processing which it becomes from step S34 - step S42 is completed, step S41 serves as Yes and shifts to step S43. #x are initialized while CPU10 increments #y in step S43 (#y<-#y+1, #x <-1). then, the step S44 -- setting -- Variable y -- TMSRT_Header of TKI#z -- it can set (TotalTMSRT_entry_Number -1) -- it judges whether the equal value was reached. # Since AOB_ELEMENT#y still reaches the last AOB_ELEMENT and y has not carried out when smaller than (TotalTMSRT_entry_Number -1), by shifting to step S32 from step S44, it continues and performs processing of step S32 - step S42. # To AOB_ELEMENT in front of [of the last AOB_ELEMENT] one, when y reaches (TotalTMSRT_entry_Number -1),

since it is thought that read-out processing of AOB_FRAME was completed, step S44 serves as Yes and shifts to step S45 of drawing 58 .

[0199] {52-9_57-2} It can be said that processing of step S45 - step S54 is the same as processing of step S33 which mentioned above two or more AOB_FRAME contained in the last AOB_ELEMENT in the point that it is the processing read, respectively - step S42. Play_data which shows the data size to which #x are "FNs_Last_TMSRTE" and reading appearance of the loop-formation processing [in / in differing / step S33 - step S42] was carried out at step S53 until now to it having been the terminating condition of loop-formation processing that #x reach "FNs_Middle_TMSRTE" in step S41 is the point that it is the terminating condition of loop-formation processing to reach SZ_DATA.

[0200] If a repeat line crack and this condition are fulfilled, step S53 will serve as Yes and processing of step S49 - step S54 will shift to step S55, until the conditions of this step S53 are fulfilled. It waits to shift to step S21, after CPU10 increments #z in step S55 (#z<-#z+1), and to accumulate the next AOB file in a double buffer 15. If accumulated, it shifts to step S22 from step S21, and about the next AOB file, processing of step S22 - step S54 will be repeated, and will be performed. That is, TKI specified by DPL_TKIN of following DPL_TK_SRP is specified and the AOB file corresponding to the TKI, i.e., the AOB file which has the same number as TKI, is specified. then, after accessing a protection field, specifying FileKey which has the same number as the TKI concerned in a cryptographic key storing file, carrying out reading appearance of the FileKey concerned and setting the FileKey concerned as the Di scrambler, reading appearance of AOB_FRAME contained in the AOB file which has the same number as the TKI is carried out one by one, and it reproduces.

[0201] {52-9_57-3_59} Updating drawing 59 of playback progress time of day is drawing in which variable Play_Time updates and the playback progress time of day displayed on the time stamp frame of a liquid crystal display 5 shows each other and signs that it increases. In this Fig. (a), although playback progress time of day is 00:00:00.000, when playback of AOB_FRAME#1 is completed, time amount length 20msec of AOB_FRAME is

added to playback progress time of day, and it is updated by 00:00:00.020. When time amount length 20msec of AOB_FRAME is added to playback progress time of day when playback of AOB_FRAME#2 was completed, and playback of AOB_FRAME#6 is completed to 00:00:00.040, it turns out that playback progress time of day is 00:00:00.120.

[0202] The above is the whole aspect of AOB_FRAME output processing. Although it is as having already stated in step S31 of this flow chart to interrupt processing of this flow chart at the time of the depression of keys other than the Play key and there being a halt key and a stop key as keys other than such a Play key is also explanation ending Also when the key for making special playback perform to a regenerative apparatus besides a halt key or a stop key is pressed, processing of the flow chart of drawing 56 , drawing 57 , and drawing 58 is interrupted, and processing according to the pressed key is performed. Henceforth, after >> key is pressed and procedure, and the halt key and stop key of CPU10 in the case of performing forward-search playback are pushed, by operating a jog dial explains the procedure of CPU10 in case a time search function is performed.

[0203] {52-10_60} Forward-search playback drawing 60 is a flow chart which shows the procedure of CPU10 in the case of performing forward-search playback. When >> key is pressed by the operator and drawing 56 , drawing 57 , step S31 of drawing 58 , step S42, and step S54 are set to Yes, this flow chart is performed by CPU10.

[0204] In step S61, CPU10 supplies from AOB_FRAME#x of AOB_ELEMENT#y to x+f(t)-1 to the Di scrambler 7. When it is the frame number which "t" is intermittent playback time amount and is equivalent to intermittent playback time amount in f (t) here, and the number of data which is equivalent to intermittent playback time amount in d (t), at step S62 play_time which shows playback progress time of day, and play_data which shows the number of reproduced data t: Intermittent playback time amount, f (t) : The frame number equivalent to intermittent playback time amount, d(t): update based on the number of data equivalent to intermittent playback time amount (x<-x+f (t) --) play_time<-play_time+t and play_data<-play_data+d (t) Still more generally intermittent playback time amount is equivalent to 240

mses (playback time amount length of 12 AOB_FRAME). .

[0205] {52-10_60-1_61A, B} Drawing 61 (a) and (b) are drawings showing signs that the increment of the playback progress time of day is carried out at the time of forward-search playback. Drawing 61 (a) shows the initial state of playback progress time of day, and shows that it is AOB_FRAME#1 of AOB_ELEMENT#51 at the playback event. It turns out that the playback progress time of day in this case is 00:00:01.000. Here, as intermittent playback time amount, AOB_FRAME from the 1st to the 12th is supplied to the Di scrambler 7, and if the 240 mses which are the time amount length of 1 AOB_FRAME are added to playback progress time of day, as shown in drawing 61 (b), playback progress time of day will be set to 00:00:01.240.

[0206] {52-10_60-2} After updating these, in step S63, a size comparison is carried out and CPU10 judges [of AOB_FRAME#x after an increment, and the total frame number of AOB_ELEMENT#y] whether AOB_FRAME#x after an increment exists in AOB_ELEMENT#y. The frame number of AOB_ELEMENT located in the head of AOB is "FNs_1 st_TMSRTE", and since the frame number of the thing of "FNs_Middle_TMSRTE" and the last is shown in "FNs_Last_TMSRTE", specifically, the frame number of a middle thing judges by comparing these with AOB_FRAME#x. If AOB_FRAME#x after updating does not exist in AOB_ELEMENT, it judges whether AOB_ELEMENT which follows AOB_ELEMENT#y in step S64 exists. Although AOB_ELEMENT#y is the last AOB_ELEMENT here, step S64 serves as No and processing of this flow chart is ended when AOB_ELEMENT which follows does not exist When AOB_ELEMENT which follows exists, it sets to step S65. The frame number in AOB_ELEMENT#y is subtracted from AOB_FRAME#x. It changes into the frame location of AOB_FRAME in (y<-y +1) and AOB_ELEMENT#y which follows AOB_FRAME#x by updating #y in step S66. If AOB_FRAME#x after updating exists in AOB_ELEMENT, these steps S65 - step S66 are skipped, and it shifts to step S67.

[0207] {52-10_60-3} Then, according to intermittent skip spacing, renewal of AOB_FRAME#x, playback progress time-of-day play_time, and number play_data of reproduced data is performed. Time amount equivalent to

intermittent skip spacing is made into skip_time (2 seconds) here. If the frame number equivalent to intermittent skip spacing skip_time is set to data severald (skip_time) equivalent to f (skip_time) and intermittent skip spacing skip_time These are used in step S67. AOB_FRAME#x, Playback progress time-of-day play_time and number play_data of reproduced data are updated (x<-x+f (skip_time), play_time<-play_time+skip_time, play_data<-play_data+d (skip_time)).

[0208] {52-10_60-4_61C} As shown in drawing 61 (c), intermittent skip spacing should be added to AOB_FRAME#x which shows the frame location in AOB_ELEMENT#51. If #x after this addition exceed the frame number of AOB_ELEMENT#51, while updating AOB_ELEMENT to following AOB_ELEMENT, AOB_FRAME#x is changed into the frame location in AOB_ELEMENT#52 from #x after addition by reducing the frame number of AOB_ELEMENT#51. In this case, AOB_ELEMENT#y is set to AOB_ELEMENT#52 and playback progress time of day is set to 00:00:03.240 by adding 2.000 to 00:00:01.240. AOB_FRAME#x is set to AOB_FRAME#62 (= (3240msec-2000msec) / 20msec) in AOB_ELEMENT#52.

[0209] {52-10_60-5_61 (d)} If AOB_FRAME#62 in AOB_ELEMENT#52 are supplied to the Di scrambler 7 after that, as shown in drawing 61 (d), playback progress time of day will be set to 00:00:03.480 by adding 0.240 to 00:00:03.240. If updating according to intermittent skip time amount is performed in step S67, step S68 - step S71 will be processed. Processing of this step S68 - step S71 is the same as processing of step S63 - step S66, if the judgment of whether AOB_FRAME after the frame number equivalent to intermittent skip spacing skip_time was added exists in AOB_ELEMENT#y is made and it exists, that following AOB_ELEMENT is made into AOB_ELEMENT#y and AOB_FRAME#x is changed into the frame location in new AOB_ELEMENT#y.

[0210] If the increment of AOB_FRAME#x and AOB_ELEMENT#y is carried out according to intermittent playback time amount and intermittent skip time amount, in step S72, CPU10 will detect AOB_FRAME#x by computing the start address about AOB_ELEMENT#y with reference to TKTMSRT, and starting retrieval of an ADTS header from the start address in

AOB_ELEMENT#y in step S73. And in step S74, after judging whether keys other than a forward direction skip key were pressed, in step S61, from AOB_FRAME#x of AOB_ELEMENT#y to x+f(t)-1 is supplied to the Di scrambler 7, and again, processing of step S62 - step S73 is repeated, and is performed.

[0211] The increment of AOB_FRAME#x and AOB_ELEMENT#y is carried out by the above processing, and a playback location advances. Then, if a playback key is pressed by the operator, step S74 will serve as No and will end processing of this flow chart.

{52-11} Processing when the activation time search function of a time search function is performed is explained. A chart example and assignment of the truck of arbitration are received for the truck in Default_Playlist information. If a truck is specified and a jog dial is operated, playback start time will be updated. If a playback key is pressed after playback start time fluctuates, the time of day when the playback was specified is specified with Jmp_Entry (second). It judges whether on the other hand, the specified truck consists of two or more AOB(s), or it consists of single AOB. When consisting of single AOB, AOB_ELEMENT#y which fills {a formula 2}, and AOB_FRAME#x are computed. If retrieval of AOB_FRAME#x is begun and it is searched for x-th AOB_FRAME from the address located in the y+2nd in TKTMSRT corresponding to this AOB if AOB_ELEMENT#y and AOB_FRAME#x which fill {a formula 2} are computed, playback will be started from this x-th AOB_FRAME.

[0212] {52-12} When consisting of two or more AOB(s), AOB#n and AOB_ELEMENT#y which fill {a formula 3}, and AOB_FRAME#x are computed. If retrieval of AOB_FRAME#x is begun and it is searched for x-th AOB_FRAME from the address located in the y+2nd in TKTMSRT corresponding to this AOB#n if AOB#n, AOB_ELEMENT#y, and AOB_FRAME#x which fill {a formula 3} are computed, playback will be started from this x-th AOB_FRAME.

[0213] Then, FNs_1 st_TMSRTE in BIT is 80 frames, and FNs_Last_TMSRTE explains the case where playback is started from the playback time of day of arbitration, in AOB 50 frames and whose FNs_Middle_TMSRTE are 94

frames.

{52-13_62A and B} -- here, as an example in case a time search function is performed, by the jog dial, when playback start time is specified, it explains how AOB_ELEMENT which should start playback, and the frame location which should start playback are pinpointed. Drawing 62 is drawing showing an example in case a time search function is performed. To be shown in drawing 62 (a) here, a regenerative apparatus should be grasped, and where a certain AOB is specified as an object for playback, revolution actuation of a jog dial should do with the thumb of the right hand, Playback start time = 00:04:40.000 (=280.00sec) should be specified. If playback start time =00:04:40.000 (=280.00sec) are applied to {a formula 2} about this AOB when BIT in TKI is the content shown in drawing 62 (b) 280sec =(FNs_1 st_TMSRTE+FNs_middle_TMSRTE-y+x) x20msec Since it is set to =(80+ 94, 148+8) x20msec, AOB_FRAME of y= 148 and x= 8 is obtained as AOB_ELEMENT#y and AOB_FRAME#x which fill {a formula 2}.

[0214] Thus, it is if the y+2nd entry addresses of AOB_ELEMENT#150 (= 148+2) are acquired from TKTMSRT and 8th AOB_FRAME to playback is started from here, since it was specified with y= 148, Playback progress time of day = playback can be started from 00:04:40.000 (=280.00sec).

Explanation of the content of processing of CPU10 when the Play key is pressed above {52-14_63_64_65} is finished. Then, the edit control program stored in ROM is explained. This edit control program is performed when the Edit key is pressed, and it shows the procedure to drawing 63 , drawing 64 , and drawing 65 . Henceforth, the content of processing of an edit control program is explained, referring to these flow charts.

[0215] {52-14_63-1} If an edit control program Edit key is pressed, the dialogue screen which shows an operator any of typical editing operation, such as deletion, division, and integration, are performed in step S101 of drawing 63 will be displayed, and it will judge after that whether the processing to a dialogue screen was specified in step S102. In actuation of a dialogue screen, a >>| key and a |<< key shall be used here as the key, i.e., the vertical cursor key, for receiving vertical cursor actuation, respectively. If deletion is specified, it will shift to the loop-formation processing which

consists of step S103 and step S104. At step S103, it judges whether the >>| key and the |<< key were pressed, and judges whether the editing key was pushed in step S104. If a >>| and |<< key is pressed, it will shift to step S105 from step S103, and the directed truck will be specified as an object for edit. On the other hand, the specified truck is deleted by performing processing shown in drawing 44 and setting TKI_BLK_ATR of TKI about the specified truck as "Unused" noting that the truck which should be deleted was specified, when the editing key was pushed.

[0216] {52-14_63-2} If integrated edit processing integrated edit is specified, it will shift to the loop-formation processing which consists of step S107 - step S109 from step S102. In the loop-formation processing which consists of step S107 - step S109, the depression of a >>| key and a |<< key and the depression of an editing key are received. If a >>|| and |<< key is pressed, it will shift to step S110 from step S107, and highlighting to the directed truck will be performed. If an editing key is pushed, step S108 will serve as Yes and will shift to step S111. At step S111, the truck directed in the cursor key is specified as an object for edit, and it shifts to the loop-formation processing which consists of step S107 - step S109 again.

[0217] If the object for edit of 2 truck eye is specified, step S109 will serve as Yes and will shift to step S112. At step S112, it judges any of Type1 and Type2 the types of AOB (when AOB is allotted also before and after that, it is AOB before and behind that) arranged at the tail and head of both trucks are by referring to BIT of TKI about the truck to precede and the truck which follows. [0218] every -- if the type of AOB became clear, it judges whether it corresponds to which arrangement pattern shown in the arrangement image of AOB of each [these] type in step S113. Drawing 32 (a) It corresponds to which arrangement pattern of - (d), and if it is clear that three Type2AOB(s) do not continue after integration, the truck preceded in step S115 and the truck which follows will be unified on one truck. That is, two or more trucks chosen as an object for these actuation are unified on one truck by performing actuation shown in drawing 46 to TKI and DPL_TK_SRP which were matched with these, and rewriting TKI_BLK_ATR of TKI. Drawing 32 (a) When it corresponds to neither of the arrangement patterns of - (d) but three AOB(s)

of Type2 continue after integration, the purport which has fear of generating of an underflow in the truck after integration in step S114 is displayed, and integrated processing is interrupted.

[0219] {52-14_64-1} If division of the division processing truck of a truck is specified, it will shift to the loop-formation processing which consists of step S116 - step S117 from step S102. In the loop-formation processing which consists of step S116 - step S117, the depression of a >>| key and a |<< key and the depression of an editing key are received. If a >>|| and |<< key is pressed, it will shift to step S118 from step S116, and the directed truck will be specified as an object for edit. If an editing key is pushed, step S117 will serve as Yes and will shift to step S119. At step S119, the truck directed in the cursor key is specified as an object for edit. Then, at step S120, playback of the truck with which division was specified is started and the depression of the Mark key is received in step S121. If the depression of the Mark key is performed, playback of a truck will be suspended and it will shift to the loop-formation processing which consists of step S122 - step S123. When the revolution actuation to a jog dial is received and revolution actuation is made to a jog dial, playback start time is made to fluctuate with the revolution actuation at step S122 in step S124. Then, it shifts to the loop-formation processing which consists of step S122 - step S123 again. It is in the condition that playback start time was fluctuated by revolution actuation, and if an editing key is pushed, it will shift to step S125 from step S123, and the playback time amount on which the editing key was pushed will be specified as a division boundary in step S125 (in addition in assignment of a division boundary, an undoing function (cancellation of edit) is possible.). Then, a truck is divided by performing processing which carried out the opinion by drawing 47 in step S126, and updating DPLI and TKI.

[0220] {52-14_65-1} If setting-out edit of the setting-out edit processing play list of play lists is specified, it will shift to the flow chart of drawing 65 . In this flow chart, Variable k is a variable for directing each truck with which playback sequence is specified with the play list set up from now on, and in the flow chart of drawing 65 , after setting 1 as the variable k of a step S131 odor lever first, it shifts to the loop-formation processing which consists of step S132 -

step S134. In the loop-formation processing which consists of step S132 - step S134, the depression of a >>| key and a |<< key, the depression of an editing key, and the depression of a stop key are received. If a >>| key and a |<< key are pressed, it will shift to step S135 from step S132, and the truck directed by the >>| key and the |<< key will be specified. If an editing key is pushed, step S133 will serve as Yes and will shift to step S136. At step S136, the truck directed in the editing key is specified as a truck which should be reproduced by the k-th. Then, in step S137, Variable k is incremented and it shifts to loop-formation processing of step S132 - step S134. By repeating such processing, sequential specification of the truck of 2 truck eye, 3 truck eye, and 4 truck eye is carried out. Thus, if a stop key is pushed where two or more trucks which should be reproduced by the newly created play list are specified, it will shift to step S138 from step S134, and the PlayList information which consists of PL_TK_SRP which specifies TKI matched with these will be generated.

[0221] (Recording device)

{66-1} An example about a recording device, then the recording device of flash memory card 31 is explained. Drawing 66 is drawing showing an example of the recording device of flash memory card 31. The communication link through the Internet is possible for the recording device in this Fig., and after the SD_Audio directory has been enciphered by the electronic music distribution, when being transmitted through a communication line, or when an audio data transport stream is distributed by the electronic music distribution, it is the general-purpose personal computer which can receive these.

[0222] {67-1} Hardware configuration drawing 67 of a recording device is drawing showing the hardware configuration of a recording device. The card connector 21 for a recording device to connect flash memory card 31 in this Fig., RAM22 and the hard disk unit 23 which stored the record control program for performing integrated control of a recording apparatus, By carrying out [voice / which was inputted from the microphone] A/D conversion, encoding A/D converter 24 which obtains PCM data, and the PCM data per unit time amount, and giving an ADTS header The ACC encoder 25 which obtains AOB_FRAME, and the scramble section 26 which

enciphers AOB_FRAME using FileKey for every AOB file, After the SD_Audio directory has been enciphered by the electronic music distribution, when being transmitted through a communication line, Or the modem equipment 27 which receives an audio data transport stream when an audio data transport stream is transmitted by the electronic music distribution through a communication line, It has CPU28 which performs integrated control in a recording apparatus, the keyboard 29 which receives the actuation from an operator, and a display 30.

[0223] {67-2} What is necessary is just to write it in the data area and protection field of flash memory card 31, when a recording apparatus receives these justly when being transmitted through a communication line, where the SD_Audio directory which should be written in a data area and a protection field by the input path RT 1 - the RT4 electronic music distribution is enciphered. however, by the SD_Audio directory, when there is nothing and the audio data transport stream itself is transmitted by the electronic music distribution, and when being inputted into a recording device in the state of PCM data and inputted into a recording device in the state of a fundamental tone, a recording device should pass four input paths shown below -- an audio data transport stream is written in flash memory card 31 -- things -- **

[0224] In case the recording apparatus in this Fig. stores an audio data transport stream in flash memory card 31, there are the input path RT 1 shown in drawing 67 , the input path RT 2, the input path RT 3, and the input path RT 4 as input path of an audio data transport stream.

{67-3} The input path RT1 input path RT 1 is an input path after the SD_Audio directory has been enciphered by the electronic music distribution, when being transmitted through a communication line, or in case an audio data transport stream is transmitted through a communication line, and AOB_FRAME contained in a transport stream in this case is enciphered using different FileKey for every thing belonging to AOB of a piece. Since the need for encryption for the second time and coding does not exist to a transport stream [finishing / encryption], a SD_Audio directory or an audio data transport stream is in the enciphered condition, and is stored in RAM22.

[0225] {67-4} The input path RT2 input path RT 2 is an input path in case

voice is inputted from a microphone. In this case, A/D conversion is made to perform the voice inputted into A/D converter 24 from the microphone, and PCM data are obtained. And AOB_FRAME is obtained by making PCM data encode to the AAC encoder 25, and giving an ADTS header. Then, the audio data with which encryption was made are obtained by using FileKey for every AOB file for the scramble section 26, and making it encipher AOB_FRAME. Then, audio data are stored in RAM22.

[0226] {67-5} The input path RT3 input path RT 3 is an input path in case the PCM data by which reading appearance was carried out from CD are inputted into equipment. Since it is inputted in the state of PCM data, the PCM data concerned are inputted into the AAC encoder 25 as it is. AOB_FRAME is obtained by making the PCM data inputted such encode to the AAC encoder 25, and giving an ADTS header. Then, the audio data with which encryption was made are obtained by using FileKey for every AOB_FRAME for the scramble section 26, and making it encipher AOB_FRAME. Then, audio data are stored in RAM22.

[0227] {67-6} The input path RT4 input path RT 4 is an input path at the time of writing the transport stream inputted in three input paths RT1, RT2, and RT3 in flash memory card 31. TKI and Default_Playlist information are generated with storing of these audio data. The functional subject of this recording device is also the record program currently recorded on ROM like the case of a regenerative apparatus. That is, processings peculiar to this operation gestalt, such as record of AOB and record of TrackManager and PlayListManager, are realized by the record program currently recorded on the hard disk unit.

[0228] {67-7_68} In the above-mentioned input paths RT1, RT2, RT3, and RT4, the procedure of the record processing in the case of writing a transport stream in flash memory card 31 is explained, referring to a flow chart after the procedure of record processing. Drawing 68 is a flow chart which shows the procedure of record processing. There are things, such as Frame_Number and Data_Size, as a variable quoted in this flow chart. Frame_Number is a variable for managing the total of AOB_FRAME recorded on the AOB file until now, and Data_Size is a variable for managing the data size of AOB_FRAME

recorded on the AOB file until now.

[0229] If this flow chart is performed, in step S200, CPU28 will create DefaultPlaylist and TrackManager, and variable #z and #w will be initialized in step S201 (z<-1, w<-1). At step S202, AOB file #z is created and it stores in the data area in flash memory card 31. In this condition, the file name of AOB file #z, a file extension child, and the first cluster number will be set to the directory entry of the SD_Audio directory in a data area. In continuing step S203, CPU28 creates TKI#z, stores it in TrackManager, and in step S204, CPU28 creates DPL_TK_SRP#w and it stores it in DefaultPlaylist information. In step S205, variable #y is initialized henceforth (y<-1), and each of Frame_Number and Data_Size is initialized in step S206 (Frame_Number<-0, Data_Size<-0).

[0230] In step S207, it judges whether the input of an audio data transport stream which should be written in AOB file #z ended CPU28. It encodes with the AAC encoder 25, and when the audio data transport stream enciphered by the scramble section 26 is stored in RAM22 one after another and needs to continue the writing of cluster data, step S207 serves as No and shifts to step S209. In step S209, CPU28 judges whether the AAC audio data for cluster size were accumulated in RAM22. When cluster data are accumulated in RAM22, step S209 serves as Yes and it shifts to step S210, and after writing the AAC audio data of the cluster size accumulated in RAM22 in flash memory card 31, it shifts to step S211. When are recording of cluster data is not completed, step S210 is skipped and it shifts to step S211. In step S211, CPU28 increments Frame_Number (Frame_Number<-Frame_Number +1), and only the data size of the AOB_FRAME increments Data_Size. After performing this updating, in step S212, Frame_Number judges whether the frame number defined as "FNs_Middle_TMSRTE" was reached. Since "FNs_Middle_TMSRTE" serves as a value according to the sampling frequency at the time of an audio data transport stream being encoded, when Frame_Number reaches "FNs_Middle_TMSRTE", although step S212 serves as Yes, step S212 is set to No and when that is not right, shifts to step S207 here. Henceforth, step S207 - step S212 are repeatedly performed until step S207 and step S212 serve as Yes.

[0231] When Frame_Number reaches "FNs_Middle_TMSRTE" and step S212 is set to Yes, after shifting to step S213 from step S212, storing Data_Size in TKTMSRT of TKI#z as TMSRT_entry#y about AOB_ELEMENT#y and incrementing #y in step S214 (y<-y +1), it judges whether in step S215, Variable y amounted to 252. It is the value which shows the total of AOB_ELEMENT which can store in AOB of a piece the value 252 here, and when Variable y does not amount to 252, it shifts to step S216. At step S216, the silent condition is continuing beyond predetermined time and it judges whether audio data reached the boundary between trucks. When a silent condition does not exist, processing of step S206 - step S215 is repeated, and is performed. When Variable y amounts to 252, or when a silent condition continues beyond predetermined time, step S215 or step S216 serves as Yes, it shifts to step S217, and increments variable #z and #w (z<-z +1, w<-w +1). Then, about #z by which the increment was carried out, processing of step S202 - step S216 is repeated, and is performed. AOB containing two or more AOB_ELEMENT is written one after another in flash memory card 31 by this repeat.

[0232] Here, since it means that the input of an audio data transport stream which should be written in AOB file #z was completed when transmission of the audio data transport stream from the AAC encoder 25, the scramble section 26, and modem equipment 27 is completed, step S207 serves as Yes and shifts to step S208. After storing in the AOB file corresponding to AOB#z the audio data which CPU28 stored Data_Size in TKTMSRT of TKI#z as TMSRT_entry#y about AOB_ELEMENT#y, and were stored in RAM22 in step S208, processing of this flow chart is ended.

[0233] Although it means that the enciphered audio data transport stream was stored in flash memory card 31 by the above processing, FileKey for canceling this encryption is stored in a protection field by the following processings. While generating FileKey from which CPU28 differs whenever coding of AOB of a piece is started in the case of the input paths RT2 and RT3, setting it as the scramble section 26 and making it encipher in the scramble section 26 by the FileKey, those FileKey(s) are stored after FileKey Entry of the cryptographic key storing file which exists in a protection field.

[0234] On the other hand, the cryptographic key storing file in which an AOB file, the file which stored TKMG, the file which stored PLMG, and FileKey from which it differs for every AOB were stored in the case of the input path RT 1 is transmitted by the provider of an electronic music distribution. CPU28 receives them, writes an AOB file, the file which stored TKMG, and the file which stored PLMG in a user data area, and writes the cryptographic key storing file which stored FileKey from which it differs for every AOB in a protection field.

[0235] Since it is enciphered in a cryptographic key different, respectively, even if the cryptographic key used for encryption is decoded and the file which stored AOB is exposed in one file according to this operation gestalt as mentioned above, AOB which can be decoded by the decode is only AOB stored in one file, and does not have the effect of what on AOB stored in other files, either. Loss when a cryptographic key is exposed can be suppressed to the minimum.

[0236] In addition, the above-mentioned operation gestalt was explained as an example of a system which can expect the best effectiveness in the actual condition. An operation change of this invention can be made in the range which does not deviate from the summary. Specifically, modification implementation as shown in the following (a) - (f) is possible.

(a) Although the gestalt of this operation explained by using a record medium as semiconductor memory (flash memory card), it is not restricted to this and can transpose to optical disks, hard disks, etc., such as DVD-RAM.

[0237] (b) Although AAC was used as music data with the gestalt of this operation, it may not be restricted to this and you may be MP3 (MPEG 1 Audio Layer 3), Dolby-AC3, DTS (Digital Theater System), etc.

(c) The very thing of the file which stored TKMG, and the file which stored PLMG is not distributed in an electronic music distribution, but it distributes with the cryptographic key storing file which stored the cryptographic key from which the information which becomes the origin of TKMG and PLMG is differed for every AOB file and AOB, and by processing the information which becomes the origin of this TKMG and PLMG in a recording device, TKMG and PLMG may be obtained and you may record on flash memory card.

[0238] (d) Although the expedient top of explanation, the recording device, and the regenerative apparatus were used as another equipment, respectively, the function of a recording device may be provided in a pocket mold regenerative apparatus, and the recording device of a personal computer mold may be made to possess the function of a regenerative apparatus. Moreover, the communication equipment which can download contents from a network besides these pocket mold regenerative apparatus and a personal computer mold recording device may be made to possess the function of these regenerative apparatus and a recording device. For example, the function of the regenerative apparatus shown in the pocket mold telephone which can access the Internet at the 1st operation gestalt, and a recording device may be made to provide, and the contents which pocket mold telephone downloaded through the wireless network may be stored in flash memory card 31 as shown in the 1st operation gestalt. Furthermore, with this operation gestalt, although the recording apparatus had modem equipment 27 for connection with the Internet, it may change to this and the terminal adopter for making connection with an ISDN circuit etc. may be provided.

[0239] (e) An execute-form program may realize the procedure explained with reference to the flow chart of drawing 55 - drawing 58 , drawing 60 , drawing 63 - drawing 65 , and drawing 68 , this may be recorded on a record medium, and you may make it the object of a negotiation and a sale. Although there are an IC card, an optical disk, a floppy (trademark) disk, etc. in such a record medium, utilization is presented with the machine program recorded on these by being installed in a general purpose computer. This general purpose computer performs the installed execute-form program serially, and realizes the function of the regenerative apparatus shown in this operation gestalt, and a recording device.

[0240] (f) Although two or more AOB(s) and two or more FileKey(s) were made to store in flash memory card 31 in this operation gestalt, one AOB and one FileKey may be made to store in flash memory card 31. Moreover, AOB may not be enciphered but AOB of an AAC method may be made to store in flash memory card 31.

(The 2nd operation gestalt)

{69-1} The whole PlayListManager 2nd operation [of a configuration] gestalt in the 2nd operation gestalt is amelioration about the semi-conductor memory card which a regenerative apparatus is made to reproduce, without overlapping and reproducing the content reproduced once. Drawing 69 is drawing showing the internal configuration of PlayListManager in the 2nd operation gestalt, and TrackManager. The configuration of Playlist Manager Information (PLMGI) which came to show clearly that the configurations of PlayListManager and TrackManager shown in this Fig. and drawing 17 differ in drawing 17 is the point clarified in drawing 69 . the configuration of this PLMGI -- setting -- especially -- it should observe -- it is PLMG_RSM_PL, and by making the resumption location of playback shown in this, it is the object which a regenerative apparatus is made to reproduce, without overlapping and reproducing the content reproduced once, and is stored in the semi-conductor memory card.

[0241] {70-1} The detail block diagram 70 of PlaylistManager information is drawing showing the detailed configuration of PlaylistManager information. As shown in drawing 70 , PlaylistManager information The PLMG_ID field which occupies even the 1st byte from the 0th byte of head, The reservation field which occupies even the 3rd byte from the 2nd byte (reserved), The solvent deasphalting_ID field which occupies even the 11th byte from the 4th byte, The VERN field which occupies even the 13th byte from the 12th byte, the PLMG_PL_Ns field which occupies even the 15th byte from the 14th byte, The PLMG_AP_PL field which occupies even the 19th byte from the 16th byte, The PLMG_RSM_PL field which occupies even the 27th byte from the 20th byte, The PLMG_APP_ATR field which occupies even the 29th byte from the 28th byte, It consists of the PLMG_FCA field which occupies even the 31st byte from the 30th byte, the TKI_Ns field which occupies even the 33rd byte from the 32nd byte, and a reservation field (reserved) which occupies even the 35th byte from the 34th byte. In this PlaylistManager information, PLMG_AP_PL and PLMG_RSM_PL serve as a chief aim of the 2nd operation gestalt.

[0242] {70-2} Referring to drawing 70 henceforth about information elements other than PLMG_AP_PL and PLMG_RSM_PL, information elements other

than PLMG_AP_PL and PLMG_RSM_PL are explained previously, and explanation about PLMG_AP_PL and PLMG_RSM_PL is given after that.
It is the character string of ISO646 and "A1" which is ID which can identify PLMGI uniquely is described by the "PLMG_ID" field.
[0243] It is the character code of ISO646 and the character string of "SD-AUDIO" which shows that Book PlayListManager is data based on SD-AUDIO specification is described by the "solvent deasphalting_ID" field.
The version number in this SD-AUDIO specification is described by the "VERN" field. The detailed bit pattern of a version number is shown in the outgoing line h71 of the broken line in drawing. The information which shows a version number is described in this detail by the field which occupies from the bit number b7 to the bit number b0. For example, when the version of Book PlayListManager is a version 0.9, "09h" is described, and when it is a version 1.0, the value of "10h" is described. Moreover, the field from the bit number b15 to the bit number b8 is secured as a reservation field for future extensions.
[0244] The number of play lists treated by PLMG and the number of the play list currently recorded on this flash memory card are described by the "PLMG_PL_Ns" field.
The application category ID which shows to which category the application stored in this flash memory card belongs is described by the "PLMG_APP_ATR" field. "01h" is described by this field when the genre of the application stored in this flash memory card is a music genre, as shown in the 1st operation gestalt. On the other hand, when the genre of the application stored in this flash memory card is karaoke software, "02h" is presentation data and it is "03h" and a leading book, "04h" is described, respectively.
[0245] Since this flash memory card stores audio data as karaoke data when "02h" is described by this application category ID, in a right channel, music and a left channel record audio data on flash memory card like voice. If audio data are recorded in this way, a regenerative apparatus can realize a karaoke performance by outputting only a right channel to a channel on either side.
[0246] The "PLMG_FCA" field is the field prepared for future extensions.
The TKI number which showed the "TKI_Ns" field to the 1st operation gestalt

is described by the integral value. In addition, TKI is described as a value to a maximum of 999. Above, the explanation about information elements other than PLMG_AP_PL and PLMG_RSM_PL is finished. Then, explanation about PLMG_AP_PL and PLMG_RSM_PL is given.

[0247] {70-3} The "PLMG_AP_PL" field is the field where the track number which should be automatically reproduced in the play list which should be automatically carried out reading appearance, and the play list concerned is described, when a regenerative apparatus is loaded with this flash memory card about PLMG_AP_PL and a player is started. The bit pattern of PLMG_AP_PL is shown within the limit pulled out by the outgoing line h72 of a broken line in drawing 70 . In this bit pattern, the bit number b8 is secured to future extensions from the bit number b26 from the bit number b31, and the bit number b15 as a reservation field.

[0248] The play list number given to the play list which should carry out reading appearance of the Playlist Number field which occupies from the bit number b7 to the bit number b0 automatically is described by in 1 to 99. The number described by this field is a number of Playlist Information (PLI) shown in the 1st operation gestalt. "0" is described when specifying a default play list as this field.

[0249] The track number of what should reproduce the Track Number field which occupies from the bit number b25 to the bit number b16 among two or more trucks where playback sequence is specified by the play list concerned by which reading appearance was carried out is described (this is a track number (Track_Number) which shows the truck shown in the 1st operation gestalt.). Although PLMG_AP_PL is an item set up by the user, when not using this, it must set all the fields mentioned above as "0."

[0250] {70-4} About PLMG_RSM_PL, "PLMG_RSM_PL" Playlist Number which shows the play list used at the time of the last playback when the AOB file stored in flash memory card is already reproduced with the regenerative apparatus, Track Number which shows the truck reproduced just before among the trucks where playback sequence was specified with the play list concerned, After the time amount of which passes since the playback start time of the truck of the track number, it consists of Playback Time which

shows whether the playback stopped. The bit pattern of PLMG_RSM_PL is shown within the limit pulled out by the outgoing line h73 of a broken line in drawing 70 . In this bit pattern the bit pattern from the bit number b31 to the bit number b0 The Playlist Number field which is the same as that of PLMG_AP_PL, and occupies from the bit number b7 to the bit number b0 The play list number given to the play list with which playback sequence was referred to immediately before is described by in 0 to 99. A track number is described by it although the Track Number field which occupies from the bit number b25 to the bit number b16 was reproduced just before among two or more trucks where playback sequence is specified by the play list concerned by which reading appearance was carried out.

[0251] Differing from the bit pattern of PLMG_AP_PL is the point that the "Playback Time" field for describing Playback Time in the field from the bit number b32 to the bit number b63 is assigned. The field from the bit number b32 to the bit number b63 is assigned to description of Playback_Time for specifying the playback halting point last in the time amount precision of a ms. In addition, when a user does not use PLMG_RSM_PL, all the fields mentioned above in PLMG_RSM_PL must be set as "0."

[0252] {70-4_71} When the flash memory card shown in setting out of PLMG_RSM_PL, then the 2nd operation gestalt at the time of transition between regenerative apparatus transfers between two or more regenerative apparatus, it explains how PLMG_AP_PL and PLMG_RSM_PL are set up. Drawing 71 is drawing showing how PLMG_AP_PL and PLMG_RSM_PL are set up, when the flash memory card shown in the 2nd operation gestalt transfers between two or more regenerative apparatus. This Fig. assumes the case where flash memory card transfers between two or more regenerative apparatus, like a general-purpose personal computer -> pocket mold regenerative-apparatus -> mount mold regenerative apparatus (these devices possess the function of the regenerative apparatus shown in the 1st operation gestalt, and a recording device.). The AOB file stored in this flash memory card is the same as that of what constitutes TrackA-TrackE and was shown in drawing 16 .

[0253] The personal computer 200 was first loaded with the flash memory

card in the 2nd operation gestalt, and after the presentation data and navigation data which were shown in the 1st operation gestalt here were recorded, PLMG_AP_PL should be set up with the personal computer 200 (here, Playlist_Number"Track_Numberwhich indicates TrackC to be 0"" 3" which shows Default_Playlist information should be set up). Then, the AOB file currently recorded on the semi-conductor memory card was reproduced in sequence, such as TrackA, TrackB, and TrackC, and when playback of TrackC which has the playback time amount of 5.5 minutes passed till 3 minutes and 31 seconds, the operator should suspend playback. in this case -- PLMG_RSM_PL -- the field -- **** -- a personal computer -- 200 -- Default_Playlist -- information -- being shown -- Playlist_Number -- " -- zero -- " -- TrackC -- being shown -- Track_Number -- " -- three -- " -- describing -- having . "00:03:31.000" the event of playback stopping with it indicates it to be that it is after [from the head of TrackC] 3-minute and 31-second progress is described by the Playback_Time field in PLMG_RSM_PL. Then, as it was removed from a personal computer 200 and shown in an arrow head my71, the pocket mold player 100 should be loaded with flash memory card.

[0254] Although the regenerative apparatus which is the pocket mold player 100 started playback in the 1st operation gestalt from AOB_FRAME of the beginning of TrackA specified for Default_Playlist information, since PLMG_AP_PL and PLMG_RSM_PL are described by PlaylistManager information, with the 2nd operation gestalt, it defines from which AOB_FRAME playback is started with reference to these. When playback with a personal computer 200 stops here, to PLMG_RSM_PL Playlist_Number"0" which shows Default_Playlist information, Since Track_Number"3" which shows TrackC, and Playback_Time which shows "00:03:31.000" are described, the pocket mold player 100 TrackC specified for Default_Playlist information can know already being reproduced till 3 minutes and 31 seconds, and can already know that what is necessary is just to reproduce TrackC from 3 minutes and 31.001 seconds.

[0255] The operator wore the headphone attached to the pocket mold player 100, and while hearing TrackC by which playback was started such, he should go out. It was reproduced in sequence, such as TrackC and TrackD, in

the meantime, and when playback of TrackD which has the playback time amount of 30.6 minutes passed till 10 minutes and 30 seconds, the operator should suspend playback. in this case -- carrying -- a mold -- a player -- 100 -- PLMG_RSM_PL -- Default_Playlist -- information -- being shown -- Playlist_Number -- " -- zero -- " -- TrackD -- being shown -- Track_Number -- " -- four -- " -- playback -- having stopped -- an event -- TrackD -- a head -- from -- ten -- a minute -- 30 -- a second -- progress -- after -- it is -- things -- being shown -- " -- 00 -- : -- ten -- : -- 30 . -- 000 -- " -- updating -- having . on the other hand -- PLMG_AP_PL -- receiving -- rewriting -- carrying out -- not having -- Default_Playlist -- information -- being shown -- Playlist_Number -- " -- zero -- " -- TrackC -- being shown -- Track_Number -- " -- three -- " -- describing -- having had -- as -- becoming .

[0256] Then, as it was removed from the pocket mold player 100 and shown in an arrow head my72, the mounted player 300 should be loaded with flash memory card. Playlist_Number"0" which shows Default_Playlist information to PLMG_RSM_PL, Since Track_Number"4" which shows TrackD, and Playback_Time which shows "00:10:30.000" are described, the mounted player 300 TrackD specified for Default_Playlist information can know already being reproduced till 10 minutes and 30 seconds, and can already know that what is necessary is just to reproduce the TrackD concerned from 10 minutes and 30.001 seconds. When TrackD was reproduced from this event and 9 minutes and 30 seconds passed after that, the operator should suspend playback. In this case, although Playlist_Number and Track_Number are the same since the non-reproduced part remains in TrackD, Playback_Time in PLMG_RSM_PL is updated by "00:20:00.000."

[0257] When flash memory card is picked out from a personal computer 200 by the above explanation and playback is started in the pocket mold player 100, the playback event of the degree at the last playback event in a personal computer 200 shows that playback is started. When similarly flash memory card is taken out from the pocket mold player 100 and playback is started in the mounted player 300, the playback event of the degree at the last playback event in the pocket mold player 100 shows that playback is started. Even when flash memory card transfers between the personal computer 200 ->

pocket mold player 100 -> mount players 300, the part reproduced by then and the overlapping part are not reproduced.

[0258] {70-5} In the place which finished the explanation about PLMG_AP_PL at the time of edit of TKI, updating PLMG_AP_PL of PLMG_RSM_PL, and PLMG_RSM_PL One truck is divided, when the truck of four edit cases part described in the 1st operation gestalt is deleted (case1) and it unifies two of two or more trucks of arbitration on one truck (case3). When obtaining two trucks (case4), and the sequence of a truck is replaced (case5), it explains how PLMG_AP_PL and PLMG_RSM_PL are updated.

[0259] When the truck specified in PLMG_AP_PL and PLMG_RSM_PL is deleted (case1), Track_Number in PLMG_AP_PL in PlaylistManager information and PLMG_RSM_PL is set as the truck located in degree ranking of the deleted truck. Moreover, about Playback_Time, it is set as the playback start time "00:00 00. 000 seconds" of the truck.

[0260] When the truck specified in PLMG_AP_PL and PLMG_RSM_PL is unified with other trucks (case3), Track_Number in PLMG_AP_PL in PlaylistManager information and PLMG_RSM_PL is updated in the ranking after the integration. When the truck specified in PLMG_AP_PL and PLMG_RSM_PL is divided (case4), Track_Number in PLMG_AP_PL in PlaylistManager information and PLMG_RSM_PL is updated by the truck of the first half in which it was obtained by the division, and the second half. That is, when a division boundary is compared with Playback_Time and Playback_Time exists ahead from a division boundary, Track_Number of the truck of the first half is set as PLMG_RSM_PL among the trucks obtained by division. When a division boundary is compared with Playback_Time and Playback_Time exists more back than a division boundary, Track_Number of the truck of the second half is set as PLMG_RSM_PL among the trucks obtained by division.

[0261] When the sequence of the truck specified in PLMG_AP_PL and PLMG_RSM_PL interchanges (case5), Track_Number in PLMG_AP_PL in PlaylistManager information and PLMG_RSM_PL is updated in the ranking after the exchange. Although the case where PLMG_AP_PL and PLMG_RSM_PL were updated was explained with edit of a truck, when

editing operation is performed, PLMG_AP_PL and PLMG_RSM_PL which were set up till then simply may be deleted.

[0262] {72-1} The regenerative apparatus in setting out, then the 2nd operation gestalt of any to refer to between PLMG_RSM_PL and PLMG_AP_PL is explained. The point that the regenerative apparatus in the 2nd operation gestalt differs from what was shown in the 1st operation gestalt is roughly divided, and there are three. The 1st difference point is a point of receiving setting out of PLMG_AP_PL, and starting setting out from an operator. Drawing 72 is drawing showing the menu screen for receiving setting out of PLMG_AP_PL, and starting setting out from an operator from an operator. In this Fig., the character string "immediately after [at the time of the last playback stopping]" and a "favorite truck" is displayed in order to receive setting out of with reference to any of PLMG_AP_PL and PLMG_RSM_PL to start playback of a truck at the time of loading of flash memory card (a "favorite truck" is a truck specified by Playlist_Number specified in PLMG_AP_PL, and Track_Number.). By setting out through this character string, a regenerative apparatus performs flag setting out. Playlist_Number this flag is described to be by PLMG_AP_PL, Playlist_Number which resumes playback from Track_Number or is described by PLMG_RSM_PL, If it is Track_Number and the flag which shows from Playback_Time whether playback is resumed (it is called a starting flag) and "immediately after [at the time of the last playback stopping]" is set up A starting flag is set as ON and playback is started from immediately after [at the time of the last playback stopping with reference to PLMG_RSM_PL at the time of loading of flash memory card].

[0263] Moreover, if a "favorite truck" is set up, a starting flag is set as OFF and can start playback from the truck indicated by PLMG_AP_PL at the time of loading of flash memory card. Moreover, in this menu screen, if setting out of a "favorite truck" is possible and alter operation is performed through a key panel, PLMG_AP_PL which shows Playlist specified by that actuation and a truck will be written in flash memory card. In addition, besides a menu screen, a DIP switch and a push button type switch may be formed in a regenerative apparatus, and a starting flag may be switched using these switches.

[0264] {56_57_58-1} More than the renewal of PLMG_RSM_PL is the 1st difference point. The 2nd difference point following this is a point of updating PLMG_RSM_PL, when a stop key is pushed during playback. When a stop key is pushed during playback in the 1st operation gestalt, in the flow chart of drawing 56 , drawing 57 , and drawing 58 , it is set to Yes any of step S31, step S42, and step S54 they are, and processing of a flow chart is ended. In this case, write-in processing of PLMG_RSM_PL shown below is performed. Namely, while specifying Playlist_Number which shows Playlist which received the stop order and was used for reference of playback sequence at the event, and Track_Number corresponding to the audio block currently reproduced and writing in PLMG_RSM_PL, a stop order is received with reference to play_time shown in the 1st operation gestalt, playback progress time-of-day Play_Time at the event is specified, and it writes in PLMG_RSM_PL by making this into Playback_Time.

[0265] In addition to the time of the depression of a stop key as well as the case at the time of the depression of a stop key, PLMG_RSM_PL may be updated at the time of the depression of a halt key. Furthermore, the PlayList information chosen when a cell residue remains and it becomes small, Track_Number, and playback progress time of day may be described to PLMG_RSM_PL. in this case, also when an operator does not push a stop key but a regenerative apparatus stops regeneration by consumption of a cell, effective PLMG_RSM_PL is written in flash memory card -- things -- **

[0266] {73-1} It continues about playback location specification processing, and the 3rd difference point is explained. Although the AOB file was reproduced with the 1st operation gestalt in the order specified in Playlist, according to the flow chart shown in drawing 73 , playback is performed from the playback location pinpointed with the 2nd operation gestalt. Henceforth, according to this flow chart, the playback location specification processing based on PLMG_AP_PL and PLMG_RSM_PL is explained.

[0267] If processing of this flow chart is started, in step S301, CPU10 will judge any shall be used between PLMG_AP_PL and PLMG_RSM_PL at the time of loading of flash memory card with reference to the starting flag set up through the menu shown in drawing 72 . When the starting flag shows

PLMG_AP_PL, it shifts to step S302 from step S301. In step S302, with reference to PLMG_AP_PL, CPU10 is specified as TKI#z which showed TKI about the truck specified in Track_Number of the play list specified in Playlist_Number described by this to the 1st operation gestalt, and starts playback from AOB file #z corresponding to this.

[0268] In step S301, when the purport over which a starting flag gives priority to PLMG_RSM_PL is shown, in step S303, PLMG_RSM_PL is read from PlaylistManager information, PLMG_RSM_PL is read in step S303, and it judges whether Playlist_Number, Track_Number, and Playback_Time which are described by PLMG_RSM_PL by which reading appearance was carried out in step S304 are just. Since it is thought that PLMG_RSM_PL is invalid when the writing of PLMG_RSM_PL is not justly performed at the time of the last playback halt, and when the abnormalities in read-out arise in the cluster which had described PLMG_RSM_PL, it shifts to step S302 from step S304, and playback is started with reference to PLMG_AP_PL.

[0269] When Playlist_Number, Track_Number, and Playback_Time are justly described by PLMG_RSM_PL, it shifts to step S305 from step S304, and judges whether the playback time amount (TKI_PB_TM) of the truck about Playback_Time described by PLMG_RSM_PL and Track_Number described by PLMG_RSM_PL is equal. If Playback_Time and TKI_PB_TM are not equal, in the truck specified in Track_Number By **, CPU10 specifies TKI specified by Track_Number in PLMG_RSM_PL as TKI#z in step S306. the non-reproduced part remains -- things -- At step S307, AOB_ELEMENT#y which should start playback, and AOB_FRAME#x which should start playback are specified from the AOB file based on Playback_Time indicated by PLMG_RSM_PL. In the 1st operation gestalt, since processing in which AOB_ELEMENT#y corresponding to the playback start time of the arbitration of a truck and AOB_FRAME#x are specified how is already explained using {formula 1} - {a formula 3}, using these formulas, it computes AOB_ELEMENT#y and AOB_FRAME#x and starts playback from AOB_FRAME#x in AOB_ELEMENT#y of AOB file #z in step S308 after that.

[0270] If Playback_Time and TKI_PB_TM are equal, step S305 serves as Yes and shifts to step S309. Although TKI_PB_TM and Playback_Time are equal,

they judge whether Track_Number in PLMG_RSM_PL and TKI_Ns indicated by PlaylistManager information are equal. Since the non-reproduced truck remains in Playlist specified in Playlist_Number when not equal, it shifts to step S311 from step S309. In step S311, the next TKI of TKI specified by Track_Number in PLMG_RSM_PL is specified as TKI#z, and playback of AOB is started from the head of AOB file #z corresponding to the TKI#z at step S312.

[0271] TKI_PB_TM and Playback_Time -- equal -- in addition -- and since it is thought that all playbacks of the play list specified in Playlist_Number of PLMG_RSM_PL were completed when Track_Number in PLMG_RSM_PL and TKI_Ns are equal, which play list is reproduced -- that assignment is received. Since it is stored as PLMG_RSM_PL in a semi-conductor memory card as a resumption location of playback how far it was reproduced at the time of the last playback, when it takes out a semi-conductor memory card from a regenerative apparatus and loads another regenerative apparatus according to this operation gestalt as mentioned above, the another regenerative apparatus concerned can start playback from immediately after [at the time of being reproduced last time / the]. Therefore, after hearing to the middle the music album which consists of TrackA-TrackE with a certain regenerative apparatus, stop playback and it sets to another regenerative apparatus. In case the music album is reproduced, with reference to PLMG_RSM_PL described at the time of a halt of the last playback, at the time of the last playback, where is playback ending and the another regenerative apparatus concerned can specify where it has not reproduced from in the time amount precision of a ms. Therefore, even if it is the case where could reproduce the music album and transition of a semi-conductor memory card arises from immediately after the already reproduced part, an operator puts up with the truck heard once, and does not hear it.

[0272] (The 3rd operation gestalt)

{74-1} DPLI_RSM_PL and the 3rd operation gestalt of PLI_RSM_PL are operation gestalten in which the termination of the playback range is made shown as a resumption location of playback, when PLI_RSM_PL and DPLI_RSM_PL are prepared in Default_Playlist information and each PlayList

information and PlayList information and Default_Playlist information are reproduced immediately before. Drawing 74 is drawing showing the Default_Playlist information for which DPLI_RSM_PL was stored in DPLGI, and the PlayList information for which PLI_RSM_PL was stored in PLGI.

[0273] It differs from PLMG_RSM_PL in that only Track_Number and Playback_Time are indicated and do not need to indicate Playlist_Number, as for DPLI_RSM_PL (PLI_RSM_PL). Moreover, if all the trucks with which playback sequence was specified for PlayList information were reproduced, it differs in that FF value which shows that a truck is completion ending is set to Track_Number in PLI_RSM_PL.

[0274] Then, the regenerative apparatus in the 3rd operation gestalt is explained. Although it is the same as that of the 2nd operation gestalt that a regenerative apparatus writes Playlist_Number of the PlayList information, Track_Number of a truck, and Playback_Time in PLMG_RSM_PL when playback of the truck specified in order of playback of PlayList information is interrupted on the way, with the 3rd operation gestalt, Track_Number of a truck and Playback_Time are written in PLI_RSM_PL of the Playlist_Number.

[0275] Moreover, through a menu, although it is the same as that of the 1st operation gestalt for assignment of which PlayList information to be possible, if the value is not written in Track_Number of the PLI_RSM_PL, and Playback_Time, with the 3rd operation gestalt, the truck with which playback sequence was specified for the PlayList information is reproduced from a head with reference to PLI_RSM_PL of the specified PlayList information. The truck with which playback sequence was specified for the PlayList information if the value was written in Track_Number of PLI_RSM_PL of the specified PlayList information and Playback_Time is reproduced according to Track_Number and Playback_Time.

[0276] {74-2_75_76} drawing 75 is drawing showing the truck sequence which consists of playback sequence specified by the play list shown in drawing 41 of the 1st operation gestalt. Moreover, drawing 76 is drawing showing how DPLI_RSM_PL of Default_Playlist information and each PlayList information is set up. Since Default_Playlist information, PLI#1, and PLI#2 were specified, respectively, the truck sequence shall be selectively reproduced, as shown in

the playback range of drawing 76 (1), the playback range (2), and the playback range (3). When Default_Playlist information, PLI#1, and PLI#2 are specified, respectively after being reproduced like the playback range (1) - playback range (3), it explains from which range playback of each truck sequence is resumed.

[0277] Since playback of a truck sequence was interrupted in the middle of playback of TrackC at the time of assignment of {74-3_75_76} Default_Playlist information, Track_Number and Playback_Time "TrackC 00:03:31.0000" which show the resumption location of playback (4) used as the termination of the playback range (1) are set as DPLI_RSM_PL of Default_Playlist information.

[0278] Since playback of a truck sequence was completed as PLI#1 was shown in the playback range (3), Track_Number of PLI_RSM_PL of PLI#1 is set as "FF." Since playback of a truck sequence was interrupted in the middle of playback of TrackA at the time of assignment of PLI#2, Track_Number and Playback_Time "TrackA 00:01:11.0000" which show the resumption location of playback (5) used as the termination of the playback range (2) are set as PLI_RSM_PL of PLI#2.

[0279] Since PLI#3 do not specify and the truck sequence has not been reproduced, Track_Number of PLI_RSM_PL of PLI#3 is set as "00." Default_Playlist information and every -- since DPLI_RSM_PL of PlayList information is set up like drawing 75 , if Default_Playlist information is specified after assignment of PLI#1, playback of the truck sequence about Default_Playlist information will be resumed from (4) just behind the playback range (1) -- things -- **

[0280] if PLI#2 are specified after specifying Default_Playlist information and reproducing all the truck sequences of Default_Playlist information, playback of the truck sequence of (5) to PLI#2 will be started just behind the playback range (2) -- things -- ** Since playback of a truck is resumed according to Track_Number and Playback_Time which are shown in ejection and this PLMG_RSM_PL in PLMG_RSM_PL according to this operation gestalt from PLGI (or DPLGI) about that play list as mentioned above when the play list which actuation is made by the operator and should be reproduced is

specified, even if it is the case where it reproduces by specifying PlayList information, playback of the duplicate content is avoidable.

[0281] In addition, in this operation gestalt, when receiving assignment of each PlayList information from an operator, as shown in drawing 49 of the 1st operation gestalt, it is desirable [the resumption of playback of Default_Playlist information and each PlayList information], since it is performed based on Track_Number and Playback_Time which were shown in DPLI_RSM_PL to receive the assignment from a user through a menu like drawing 77 rather than to only to perform the list display of a play list. Drawing 77 is drawing showing an example of the menu screen which matched and displayed the content of setting out of DPLI_RSM_PL at the time of the playback range (1) - playback range (3) being reproduced on each play list. About the PlayList information which has the truck whose playback is not completed among PlayList information, the track number specified in Track_Number of DPLI_RSM_PL and the playback time of day based on Playback_Time are displayed. Since Track_Number of DPLI_RSM_PL is set as FF value about the PlayList information which all playbacks of the specified truck completed, based on this, the purport which all playbacks of PlayList information completed is displayed. If such a menu screen is referred to, an operator can remember easily how far each play list was heard. Moreover, which being the play list which playback of a truck sequence completed, and playback of a truck sequence can know like PLI#1 which is a play list in the sheep.

[0282] (The 4th operation gestalt) Although the 1st operation gestalt - 3rd operation gestalt stored music application in flash memory card 31, it is related with the amelioration in the case of storing transient application in flash memory card 31. Once transient application hears news, the speech in a public performance, a journal, etc., it is enough, the application which is not the thing of the property repeatedly heard like music application is said, and the thing of every day newest [thing / newest every week and every month] is sent [news] especially about a journal.

[0283] When downloading the transient application which requires a recording device through a network, while storing in flash memory card 31 the audio

data which constitute transient application as two or more AOB(s), two or more TKI(s) about each AOB are generated, and it stores in flash memory card 31. Moreover, the PlayList information which specifies TKI about transient application is also generated, and it stores in flash memory card 31 like TKI.

[0284] Then, the Default_Playlist information in the 4th operation gestalt, PlayList information, and the improving point about TKI are explained. Although PLI_APP_ATR was prepared in PlayListManager with the 2nd operation gestalt as information which shows the attribute of application, PLI_APP_ATR and TKI_APP_ATR are prepared in DPLGI, PLGI, and TKGI with the 4th operation gestalt as information which shows the attribute of application. Drawing 78 is drawing showing the data format of DPLGI, PLGI, and TKGI concerning the 4th operation gestalt. "PLMG_APP_ATR" which showed PLI_APP_ATR in PLGI to the 2nd operation gestalt -- the application category ID which shows similarly to which category PlayList information belongs is described. "01h" is described by this field when the genre of the application stored in this flash memory card is a music genre. On the other hand, when the genre of the application stored in this flash memory card is karaoke software, "02h" is presentation data and it is "03h" and a leading book, "04h" is described, respectively. When it is other genres, other values are described, respectively. The transient PlayList information about application is set as "04h" PLI_APP_ATR in PLGI indicates a leading book to be.

[0285] Thus, the PlayList information about transient application is generated, transient application is matched with this, and it stores in flash memory card 31. Then, the trouble produced in case transient application is accumulated in flash memory card 31 is explained. Since the thing newest every day is sent, if the transient application of a news genre accumulates this in flash memory card 31 serially as for it, the storage capacity to which flash memory card 31 was restricted will be instantly occupied by only transient application.

[0286] In order to prevent occupancy by such transient application, it is desirable to perform the following control with reference to PLI_RSM_PL and PLI_APP_ATR. Since the PlayList information to which PLI_APP_ATR was

set with the leading book genre is memorized by flash memory card 31, if this transient application refers to this PLI_APP_ATR, it will become clear which are the PlayList information corresponding to transient application, and TKI and AOB. About the PlayList information which is the PlayList information set as transient application on the other hand, and playback of the truck with which playback sequence is specified has completed Track_Number in PLI_RSM_PL about PlayList information is set as the value of FF value. About the incomplete thing of playback of the truck with which playback sequence is specified Since Track_Number in PLI_RSM_PL about Default_Playlist information is set up in addition to FF value When Track_Number in DPLI_RSM_PL confirms whether to be FF value, it can judge whether playback of transient application is completed, or it is incomplete. If TKI and AOB about the transient application which playback of a truck has completed, and PlayList information are eliminated after passing through such a check, it is avoidable that storing of flash memory card 31 receives pressure by are recording of much transient applications. In addition, although explanation of the above control was given for PLI_RSM_PL and PLI_APP_ATR, same control may be performed about DPLI_RSM_PL and DPLI_APP_ATR.

[0287] Even if transient application is transmitted like every day sequentially from the thing which had these transient applications reproduced according to this operation gestalt as mentioned above since transient application is eliminated when transient applications, such as news, are downloaded and it stores in flash memory card 31, flash memory card 31 can prevent being occupied only by transient application.

[0288]

[Effect of the Invention] Since the semi-conductor memory card concerning this invention is characterized by storing the audio sequence which comes to arrange two or more audio objects, and the resume information which shows the restart location in the case of resuming playback from the middle of an audio sequence, it presupposes that an audio sequence corresponds to a music album and was heard to the middle in the regenerative apparatus with this audio sequence. Then, supposing another regenerative apparatus is loaded with that semi-conductor memory card, this another regenerative

apparatus will resume playback of an audio sequence from the resumption location of playback shown in resume information.

[0289] Since the resumption of playback based on resume information is made even if actuation of what is not made by the operator, when another regenerative apparatus is loaded with a semi-conductor memory card, it does not trouble the time and effort of making an operator specify the music (audio object) which should be reproduced in the another regenerative apparatus. Here, said resume information may show the 1st restart location where the 1st positional information was set up through user operation including the 1st positional information and/or the 2nd positional information, and the 2nd positional information may show the 2nd restart location automatically set up at the time of the last playback halt.

[0290] Each of two or more audio objects which can set the 2nd object to said audio sequence The identification information of a proper is given. Said 1st positional information The identification information given to any one audio object shows the 1st restart location in an audio sequence. Said 2nd positional information The identification information of any one audio object and the hour entry which shows the offset from the head of the audio object to the 2nd restart location may show the 2nd restart location in an audio sequence. In the 2nd positional information, since the offset from the head of an audio object is shown, when transition between regenerative apparatus arises, the regenerative apparatus concerned can start playback from immediately after [at the time of being reproduced last time]. Therefore, even if it stops playback, and the another regenerative apparatus concerned is the case where could reproduce the music album and transition of a semi-conductor memory card arises from immediately after the already reproduced part in case the music album is reproduced in another regenerative apparatus after hearing a music album to the middle with a certain regenerative apparatus, an operator puts up with the music listened to once, and does not listen to it.

[0291] Even if it is the case where hearing the music album heard with a certain regenerative apparatus with another regenerative apparatus like [in the case of viewing and listening to the music album acquired in the electronic

music distribution] arises frequently, as for another regenerative apparatus, duplication playback of a reproduced part is avoidable.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is drawing showing the configuration at the time of seeing flash memory card 31 from a top face.

[Drawing 2] It is drawing showing the configuration at the time of seeing flash memory card 31 from the underside.

[Drawing 3] It is drawing showing the layered structure of the flash memory card 31 concerning this operation gestalt.

[Drawing 4] (a) It is drawing showing the configuration of the system area established in the physical layer of flash memory card 31, a protection field, and a user data area. (b) It is drawing showing the configuration of the protection field in a file system layer, and a user data area.

[Drawing 5] It is drawing showing the detail of the configuration in a file system layer.

[Drawing 6] It is drawing supposing the condition of dividing AOB001.SA1 into five in all at cluster size, and storing each division part in Clusters 003, 004, 005, 00A, and 00C.

[Drawing 7] It is drawing showing a directory entry in case AOB001.SA1 is recorded on two or more clusters, and the example of setting out about a file allocation table.

[Drawing 8] (a) When it stores these [in the (b) application layer] two data, it is drawing showing what kind of file is created by the subordinate of the directory concerned by what kind of directory being constituted by a user data area and the protection field in a file system layer.

[Drawing 9] They are AOBSA1.KEY under an SD_Audio directory, and drawing showing a response with an AOB file.

[Drawing 10] It is drawing showing the data configuration of an AOB file hierarchical.

[Drawing 11] (a) It is drawing showing the parameter described by ISO/IEC 13818-7 in a tabular format.

(b) It is drawing showing the parameter which should be used in case it encodes by MPEG-Layer3 (MP3) method in a tabular format.

(c) It is drawing showing the parameter which should be used in case it encodes by the Windows Media Audio (WMA) method in a tabular format.

[Drawing 12] It is drawing showing the detail of the configuration of AOB_FRAME.

[Drawing 13] In three AOB_FRAME, it is drawing showing how the cutting tool length of the audio data in each AOB_FRAME is set up.

[Drawing 14] It is drawing showing a response with sampling_frequency and the AOB_FRAME number contained in AOB_ELEMENT.

[Drawing 15] It is drawing showing an example of the time amount length of AOB_ELEMENT, and the time amount length of AOB_FRAME.

[Drawing 16] It is drawing showing what kind of content of playback is reproduced by reproducing continuously each AOB and AOB_BLOCK which are recorded on the AOB file.

[Drawing 17] It is drawing which detailed gradually the configuration of Playlistmanager in the 1st operation gestalt, and TrackManager.

[Drawing 18] It is drawing showing the size of PlayListManager and TrackManager.

[Drawing 19] It is drawing showing the correlation of TKI shown in drawing 17 , and the AOB file shown in drawing 16 and AOB.

[Drawing 20] It is drawing showing the detailed DS of TKTMSRT shown in drawing 17 .

[Drawing 21] It is drawing showing an example about TKTMSRT.

[Drawing 22] It is drawing showing the detail configuration of TKGI.

[Drawing 23] (a) It is drawing showing the detail configuration of (b) BIT.

(c) It is drawing showing the data format of the TIME_LENGTH field.

[Drawing 24] It is drawing showing the cluster 007 in which AOB which consists of AOB_ELEMENT#1-#4 is stored - cluster 00E.

[Drawing 25] When performing forward-search playback from AOB_FRAME#x in AOB_ELEMENT#y of the arbitration in AOB, it is drawing showing how AOB_FRAME#x +1 which should be reproduced next is set up.

[Drawing 26] (a) When the playback start time of (b) arbitration is specified, it is drawing showing how AOB corresponding to the appointed time of day, AOB_ELEMENT, and AOB_FRAME are specified.

[Drawing 27] (a) It is drawing supposing the case where the (b) truck is deleted.

[Drawing 28] (a) It is drawing showing TrackManager after deletion of a truck was performed two or more times.

(b) When TKI of "Unused" exists and it writes in new TKI and an AOB file here, it is drawing showing how the writing is performed.

[Drawing 29] (a) When unifying a (b)2 ** truck, it is drawing showing how TKI is set up.

[Drawing 30] (a) It is drawing showing AOB of Type1.

(b) It is drawing showing AOB of Type2.

[Drawing 31] (a) It is drawing showing the case where a multiple track is unified to one, in the combination of Type1+Type2+Type2+Type1.

(b) It is drawing showing the case where a multiple track is unified to one, in the combination of Type1+Type2+Type2+Type2+Type1.

[Drawing 32] (a) It is drawing showing the arrangement pattern with which AOB of Type1 is allotted to the termination of the truck to precede, and AOB of Type1 is allotted to the head of the truck which follows.

(b) It is drawing showing the arrangement pattern with which AOB of Type1 is allotted to the termination of the truck to precede, and AOB of Type2 is allotted to the head of the truck which follows.

(c) It is drawing showing the arrangement pattern with which AOB is allotted to the termination of the truck to precede in Type1 and Type2 order, and AOB of Type1 is allotted to the head of the truck which follows.

(d) It is drawing showing the arrangement pattern with which AOB is allotted to the termination of the truck to precede in Type1 and Type2 order, and AOB of Type2 and Type1 is allotted to the head of the truck which follows.

(e) It is drawing showing the arrangement pattern with which AOB of Type2

and Type2 is allotted to the termination of the truck to precede, and AOB of Type1 is allotted to the head of the truck which follows.

[Drawing 33] (a) It is drawing supposing the case where a (b)1 ** truck is divided into two trucks.

[Drawing 34] (a) -- or [that the SD_Audio directory entry about the SD_Audio directory where AOB003.SA1 belongs before and after (b) division is described how] -- it is drawing showing **.

[Drawing 35] (a) It is drawing supposing the case where AOB is divided in a part in the middle of AOB_ELEMENT#2.

(b) It is drawing showing the condition that AOB was divided in the part in the middle of AOB_ELEMENT#2, and two AOB(s) called AOB#1 and AOB#2 were obtained.

[Drawing 36] As shown in drawing 35 , when AOB is divided, it is drawing showing how BIT is set up.

[Drawing 37] It is drawing showing concretely how BIT changes before and after division.

[Drawing 38] It is drawing showing concretely how TKTMSRT changes before and after division.

[Drawing 39] (a) It is drawing showing a format of DPL_TK_SRP.

(b) It is drawing showing a format of PL_TK_SRP.

[Drawing 40] It is drawing showing the correlation of Default_Playlist information, TKI, and an AOB file.

[Drawing 41] It is drawing having shown the example of setting out of DefaultPlaylist and PlayList information with the same notation as drawing 40 .

[Drawing 42] It is drawing showing a response with DPL_TK_SRP and TKI using the same notation as drawing 40 .

[Drawing 43] (a) It is drawing supposing the case where the sequence of the (b) truck is replaced.

[Drawing 44] (a) When deleting DPL_TK_SRP#2 and TKI#2 among DefaultPlaylist(s) shown in (b) drawing 40 , it is drawing showing how DefaultPlaylist, TrackManager, and an AOB file are updated.

[Drawing 45] (a) When TKI of (b "Unused") and DPL_TK_SRP exist and it writes in new TKI and DPL_TK_SRP here, it is drawing showing how the

writing is performed.

[Drawing 46] (a) It is drawing supposing the case where the (b) truck is unified.

[Drawing 47] (a) It is drawing supposing the case where the (b) truck is divided.

[Drawing 48] It is drawing showing the regenerative apparatus of the pocket mold about the flash memory card 31 concerning this operation gestalt.

[Drawing 49] It is drawing showing an example of the content of a display of the liquid crystal display at the time of selection of a play list being performed.

[Drawing 50] (a) It is drawing showing an example of the content of a display of the liquid crystal display at the time of selection of - (e) truck being performed.

[Drawing 51] (a) It is drawing showing the example of actuation of -(c) jog dial.

[Drawing 52] It is drawing showing the internal configuration of a regenerative apparatus.

[Drawing 53] It is drawing showing how the data I/O in a double buffer 15 is performed.

[Drawing 54] (a) It is drawing showing how partitioning of the patrol type using (b) ring pointer is performed.

[Drawing 55] It is the flow chart which shows the procedure of AOB file read-out processing.

[Drawing 56] It is the flow chart which shows the procedure of AOB_FRAME output processing.

[Drawing 57] It is the flow chart which shows the procedure of AOB_FRAME output processing.

[Drawing 58] It is the flow chart which shows the procedure of AOB_FRAME output processing.

[Drawing 59] (a) The playback progress time of day displayed on the time stamp frame of the - (d) liquid crystal display 5 is drawing in which variable Play_Time's updating and showing each other and signs that it increases.

[Drawing 60] It is the flow chart which shows the procedure of CPU10 at the time of forward-search regeneration.

[Drawing 61] (a) It is drawing showing signs that the increment of the playback progress time of day is carried out at the time of -(d) forward-search playback.

[Drawing 62] (a) It is drawing showing an example in case -(b) time search function is performed.

[Drawing 63] It is the flow chart which shows the procedure of an edit control program.

[Drawing 64] It is the flow chart which shows the procedure of an edit control program.

[Drawing 65] It is the flow chart which shows the procedure of an edit control program.

[Drawing 66] It is drawing showing an example of the recording device of flash memory card 31.

[Drawing 67] It is drawing showing the hardware configuration of a recording device.

[Drawing 68] It is the flow chart which shows the procedure of record processing.

[Drawing 69] It is drawing showing the internal configuration of PlayListManager in the 2nd operation gestalt, and TrackManager.

[Drawing 70] It is drawing showing the detailed configuration of PlaylistManager information.

[Drawing 71] When the flash memory card shown in the 2nd operation gestalt transfers between two or more regenerative apparatus, it is drawing showing how PLMG_AP_PL and PLMG_RSM_PL are set up.

[Drawing 72] It is drawing showing the menu screen for receiving setting out of PLMG_AP_PL, and starting setting out from an operator from an operator.

[Drawing 73] It is the flow chart which shows the procedure of the playback location specification processing based on PLMG_AP_PL and PLMG_RSM_PL.

[Drawing 74] It is drawing showing the DS at the time of storing 6 bytes of high order of PLMG_RSM_PL in DPLGI about Default_Playlist information, and PLGI about PlayList information.

[Drawing 75] It is drawing showing how DPLI_RSM_PL of Default_Playlist information and each PlayList information is set up.

[Drawing 76] It is drawing showing the truck sequence which consists of playback sequence specified by the play list shown in drawing 41 of the 1st

operation gestalt.

[Drawing 77] It is drawing showing an example of the menu screen which matched and displayed the content of setting out of DPLI_RSM_PL at the time of the playback range (1) - playback range (3) being reproduced on each play list.

[Drawing 78] It is drawing showing the data format of DPLGI, PLGI, and TKGI concerning the 4th operation gestalt.

[Description of Notations]

1 Card Connector

2 User Interface Section

3 RAM

4 ROM

5 Liquid Crystal Display

6 LCD Driver

7 Di Scrambler

8 AAC Decoder

9 A/D Converter

11 DPLI Resident Area

12 PLI Storing Field

13 TKI Storing Field

14 FileKey Storing Field

15 Double Buffer

21 Card Connector

22 RAM

23 Hard Disk Unit

24 Converter

24 A/D Step S

25 AAC Encoder

26 Scramble Section

27 Modem Equipment

28 CPU

29 Keyboard

30 Display

31 Flash Memory Card

32 Protection Switch

100 Pocket Mold Player

200 Personal Computer

300 Mounted Player